ACCOUNTING FOR PARAMETER UNCERTAINTY IN REDUCED-ORDER STATIC AND DYNAMIC SYSTEMS

A Dissertation

by

DREW PATTON WOODBURY

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2011

Major Subject: Aerospace Engineering

Accounting for Parameter Uncertainty in

Reduced-order Static and Dynamic Systems

ACCOUNTING FOR PARAMETER UNCERTAINTY IN REDUCED-ORDER STATIC AND DYNAMIC SYSTEMS

A Dissertation

by

DREW PATTON WOODBURY

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	John L. Junkins
Committee Members,	Daniele Mortari
	Thomas Pollock
	Alexander Parlos
	John Crassidis
Head of Department,	Dimitris Lagoudas

December 2011

Major Subject: Aerospace Engineering

ABSTRACT

Accounting for Parameter Uncertainty in

Reduced-order Static and Dynamic Systems. (December 2011)

Drew Patton Woodbury, B.S., Massachusetts Institute of Technology

Chair of Advisory Committee: Dr. John L. Junkins

Parametric uncertainty is one of many possible causes of divergence for the Kalman filter. Frequently, state estimation errors caused by imperfect model parameters are reduced by including the uncertain parameters as states (i.e., augmenting the state vector). For many situations, this not only improves the state estimates, but also improves the accuracy and precision of the parameters themselves. Unfortunately, not all filters benefit from this augmentation due to computational restrictions or because the parameters are poorly observable. A parameter with low observability (e.g., a set of high order gravity coefficients, a set of camera offsets, lens calibration controls, etc.) may not acquire enough measurements along a particular trajectory to improve the parameter's accuracy, which can cause detrimental effects in the performance of the augmented filter. The problem is then how to reduce the dimension of the augmented state vector while minimizing information loss.

This dissertation explored possible implementations of reduced-order filters which decrease computational loads while also minimizing state estimation errors. A theoretically rigorous approach using the "consider" methodology was taken first to static, or algebraic, systems and then the theory was expanded to include dynamic models. Methods for discrete, continuous, and continuous dynamic system models measured

at discrete time intervals were explored for linear systems. The continuous dynamics, discretely measured (continuous-discrete) model was also expanded for use with nonlinear systems. Additional techniques for reduced-order filtering are presented including the use of additive process noise, an alternative consider derivation, and the minimum variance reduced-order filter. Multiple simulation examples are provided to help explain critical concepts. Finally, two hardware applications are also included to show the validity of the theory for real world applications.

It was shown that the minimum variance consider Kalman filter (MVCKF) is the best reduced-order filter to date both theoretically and through hardware and software applications. The consider method of estimation provides a compromise between ignoring parameter error and completely accounting for it in a probabilistic sense. Based on multiple measures of optimality, the consider filtering framework can be used to account for parameter error without directly estimating any or all of the parameters. Furthermore, by accounting for the parameter error, the consider approach provides a rigorous path to improve state estimation through the reduction of both state estimation error and with a consistent variance estimate. While using the augmented state vector to estimate both states and parameters may further improve those estimates, the consider estimation framework is an attractive alternative for complex and computationally intensive systems, and provides a well justified path for parameter order reduction.

To My Parents: Ben and Eileen Woodbury

ACKNOWLEDGEMENTS

Due to the amount of effort required to complete a doctoral degree, no one person could finish successfully without the support of friends and family. I would first like to thank my parents, Ben and Eileen, for supporting and believing in me, as well as helping me find the courage to return to school. I'd also like to thank my close friends Justin "Jack" Jackson and Ryan Weisman, who have provided me with great insight over the years with my graduate studies and beyond.

I want to thank Dr. Manoranjan Majji for first introducing me to this area of study and helping me get started with the preliminary research and derivations. Dr. Jeremy Davis and James Doebbler brought my research to a new level by helping me focus not only on the theory, but also on its application. Dr. Julie Parish helped immensely in the setup and operation as well as providing the theoretical background for the Stellar Positioning System. Special thanks also goes out to Kurt Cavalieri of LASR Laboratory, who assisted in the setup of the odometry problem and helped gather the required data from HOMER. The help of Dr. Bong Su Koh is also appreciated for helping me fine tune my research results.

Dr. John Junkins has been an amazing advisor by allowing me to chase after my own ideas and insights and all the while guiding me towards becoming a better engineer and man, in general. Dr. Daniele Mortari served as a constant source of inspiration early on in my graduate career, and continuously pushed me to expand the breadth of my research interests. I'd also like to thank Dr. Thomas Pollock, Dr.

Alexander Parlos, and Dr. John Crassidis for helping me further refine my doctoral research.

Last, but certainly not least, I want to thank the unsung heroines of my research, Lisa Willingham and Karen Knabe, who not only kept me in line, but also would constantly remind me to smile.

TABLE OF CONTENTS

Page

CHAPTER Page

LIST OF FIGURES

FIGURE Page

FIGURE Page

FIGURE Page

LIST OF TABLES

TABLE Page

CHAPTER I

INTRODUCTION

Since its inception, the Kalman filter has been used extensively in a wide range of applications and has become ubiquitous in the estimation community. Soon after its discovery, however, researchers found a myriad of situations in which the Kalman filter diverged from the optimal estimate in practical applications. Sources of divergence included but were not limited to: overly simplified system models, parameter errors, incorrect initial covariance estimates, sparse measurements, low observability, and computer round-off. This dissertation focuses on resolving divergence in the Kalman filter due to errors in the system model parameters.

The most logical solution to reducing errors caused by model parameters is to augment the state vector by including the uncertain parameters as states. For many situations this not only improves the state estimates, but also improves the accuracy and precision of the parameters themselves. Unfortunately, not all filters benefit from this augmentation due to computational restrictions or because the parameters are poorly observable. A parameter with low observability (e.g., a high-order gravity coefficient, camera offset, etc.) may not acquire enough measurements along a particular trajectory to improve its accuracy, which can cause detrimental effects on the performance of the augmented filter. In many cases, the effects of these parameters need to be included in the model for accuracy concerns, but the additional degrees of freedom are a challenge for the numerical calculations. Thus, the problem is then how to reduce the dimension of the augmented state vector while minimizing information loss.

This dissertation follows the style of *Journal of Guidance, Control, and Dynamics*.

A. Literature Review

During the technological explosion in the 1960s, multiple ideas were proposed on how to reduce the size and complexity of large scale models to fit on-board systems with very limited computing power. One of the common techniques of the time was to perform sensitivity analyses on the models to determine which states were critical and which ones could be neglected. Gelb describes this as suboptimal filtering where states can be decoupled or even deleted from the state vector to reduce the overall computation load [1]. Decoupled states are states that still need to be estimated, but their covariance with other states can be neglected due to small correlation values. Deleted states, or ignored states, are removed from both the state vector and covariance equations completely. Gelb goes on further to explain how to perform the sensitivity analysis for simple systems as well as how the calculated filter gains can be approximated.

Although Gelb provides the how-to for a sensitivity analysis, other researchers provide the formal technical foundations, or the why, as to how these methods work. Multiple authors including Huddle and Wismer [2], Mahalanabis [3], and Price [4] initially explored the effects of mismodeled parameters in the dynamic model only. Duiven expanded this work by also including sensitivity analysis for mismodeled parameters in the measurement model [5]. Nishimura [6] and Heffes [7] took a different approach by comparing the true, computed, and the actual covariances of the initial estimated state, measurement, and process noise covariances. A more recent summary of these results can also be found in the textbook written by Tapley *et al.* [8]. Nishimura went on to combine both dynamic model and measurement model sensitivity approaches to make one massive sensitivity analysis possible [9].

Even though the sensitivity analysis can provide a large amount of information regarding the importance of each state or parameter, it is a brute force approach and must be performed for each individual system. This results in large amounts of time and resources, especially for complex models, to effectively evaluate each system. Additional tuning may also be required once the sensitivities have been determined to meet specifications for the lower order system. Sensitivity analyses still play an important role in modern day system evaluation, but researchers were also interested in finding theoretical alternatives that could be quickly applied to a wide range of systems, thus, minimizing the amount of application specific heuristics and testing.

Today, the most common approach to reduce the dimensionality of a high-order system model and prevent divergence is to include or increase the process noise in the dynamic model. An additive process noise term in the dynamic model inflates the propagated state covariance because of known modeling assumptions [1]. Implicitly the covariance of the process noise "tells" the simulation about the resulting state uncertainty as a function of time. Algorithms which use the propagated state and covariance can then account for dynamic model uncertainty. Although this method frequently corrects or greatly reduces the problem, it often results in a monotonous and tedious trial-and-error tuning session of the process noise covariance matrix. The theoretical underpinning for process noise is usually the assumption of a white noise process, whereas parametric model errors are typically non-white, so there is an issue of mismatch between physical reality and the modeling assumptions in the derivation of the Kalman filter. Furthermore, incorrect choices of the process noise covariance matrix can overestimate or underestimate the state error covariance causing, larger variability in the state estimates. Similar techniques can also be applied to the measurement covariance for modeling uncertainty in the measurement equation, but the same limitations as the inflation of process noise are encountered here as well.

Other methods of model reduction were also developed since many viewed the application of process noise as an *ad hoc* or improvisation method [1, 10]. These simpler theoretical techniques are especially effective when prior knowledge of the effectiveness of certain parameters is available. Furthermore, ways to include this *a priori* knowledge were attempted while at the same time reducing computational overhead. One of the methods developed is based on controller observer theory and uses a direct reduction of the state vector via a transformation matrix. Aoki and Huddle provided an early development of this method and show how it relates to observer theory [11]. The method was further refined and was ultimately referred to as Reduced-Order Filtering (ROF) or Minimum Variance Reduced-Order Filtering (MVROF). Hutchinson *et al.* give a good summary of its derivation [12]. Maybeck also relates the use of the MVROF as means to estimate process noise values [10]. Unfortunately, the MVROF has significant drawbacks which are discussed in further detail in Chapter IV.

A more modern method for modeling errors in the measurement equation involves the use of Total Least Squares. Similar to augmenting a filter, this method is able to improve the state estimates by minimizing errors from both measurement noise as well as parameter uncertainties in the measurement model [13]. This method has even been applied to Kalman filter applications [14]. Unfortunately, the Total Least Squares approach requires a singular value decomposition (SVD) which is more computationally expensive than competing approaches. Methods have been developed to reduce the computational overhead of the SVD, but computational costs remain large compared to classical least squares techniques [15,16]. Thus, despite the improved error reduction of Total Least Squares, the SVD required for this method is computationally restrictive and goes against the initial goals of these developments.

Another method of model reduction is to "consider" the parameters. In the consider methodology, the full state vector is reduced whereby certain parameters have been removed. Different from completely ignoring them, however, their covariance and cross-covariance with the states is included (i.e., considered) to adjust the state update and propagation equations. By accounting for the error in the parameter estimates through its associated covariance, the error in the estimated states is reduced and the accuracy of the state covariance is improved. Bierman describes two methods of applying the consider methodology to a Kalman filter [17]. The first is called consider analysis where the estimated states are filtered using the traditional Kalman filter structure. The consider analysis is then performed as an extra step after each propagation and update to produce the consider states. The consider states themselves are not filtered, but stored separately from the filter structure and provide a quantifiable measure of deviations caused by incorrect parameters. Unfortunately, if the traditional Kalman filter diverges or fails, nothing is gained from consider analysis. Tapley *et al.* discuss both batch and sequential filtering methods based on the consider analysis approach [8].

Alternatively, the consider states can be filtered directly which is called consider Kalman filtering (CKF). The CKF was first developed by S.F. Schmidt in the 1960s [18]. His formulation is based on a minimum variance approach, but his original paper does not go into detail as to how to derive the result. He does, however, present results where the dynamic model and measurement model parameters are separated. The CKF is also often called the Schmidt-Kalman filter after its developer. Additional authors extended the work by providing a derivation which includes process noise [19] and deriving square-root factorization methods for the CKF [17, 20]. Jazwinski additionally presents a linearized and extended form of the CKF for applications to nonlinear systems [19]. Schlee *et al.* compared two forms of the CKF to a traditional

Kalman filter using process noise and found that all three filters produced comparable results [21]. Markley and Carpenter have also expanded the use of the CKF for use in additional sensitivity analyses by separating out errors from the initial estimates of the state, measurement, and process noise covariance matrices [22]. The CKF has also been expanded for use with unscented and other nonlinear filtering methods for use in bias approximations, formation flying, and Simultaneous Location and Mapping (SLAM) applications [23–25].

B. Dissertation Organization

The purpose of this dissertation is to find the best reduced-order filter and document its results when applied to simple simulations and hardware applications. The focus is placed on the consider Kalman filter because, as is demonstrated, this method not only generates unbiased state estimates, but also converges to a consistent covariance estimate while preventing divergence caused by parameter uncertainty. The theoretical framework for the consider methodology is applied to multiple measures of optimality and is applied to linear and nonlinear systems using both batch and sequential estimation methods. Additional reduced-order filters are also tested and evaluated. The systematic approach for developing the various "consider" and related reduced-order filters is felt to be a useful contribution of this dissertation. The main objectives of the research leading to this dissertation are:

- To establish a theoretically rigorous framework for the development of reduced-order filters that accurately capture covariance information
- To develop near optimal methods of reduced-order filtering that reduce computational cost when compared to complex higher-order state models

- To assess capabilities of reduced-order filtering as applied to parameters with low observability
- To remove inconsistencies in and compare multiple reduced-order filters found in the technical literature
- To evaluate reduced-order filters using multiple simulation examples and hardware applications

This dissertation discusses the results of this research in the following organization:

In Chapter II, basic theoretical developments are made using reduced-order estimators for static systems. Least squares, minimum variance, maximum likelihood, and Bayesian measures of optimality are derived and analyzed. Multiple techniques are developed dependent on the *a priori* information available. Both batch and sequential/recursive methods of estimation are explored.

In Chapter III, the consider methodology is expanded for use in dynamic systems with process noise. Derivations of the consider Kalman filter are presented for discrete, continuous, and continuous-discrete time systems. Stability analyses are performed for both the discrete and continuous system derivations. Linearization procedures to extend the consider Kalman filter to nonlinear systems is provided.

In Chapter IV, other methods of accounting for parameter uncertainty are presented and compared to the consider Kalman filter. Process noise is used in conjunction with a traditional Kalman filter to compare the consider Kalman filter to the most common form of accounting for uncertainty, and in order to make certain observations on theoretical and actual performance of the estimation algorithms. An alternate consider Kalman filter derivation is provided using an augmented measurement vector instead of a minimum variance application. The additional consider

Kalman filter derivation and the minimum variance reduced-order filter described above are compared to the minimum variance consider Kalman filter to assess each filter's capabilities and limitations.

In Chapter V, the consider Kalman filter is applied to two hardware applications to evaluate the filter's effectiveness on real world systems. A simple linear hardware example is used first to directly relate hardware results to previous simulations. A more complex attitude application is then used to demonstrate the consider Kalman filter's capabilities.

CHAPTER II

PARAMETER UNCERTAINTY IN STATIC SYSTEMS

In static systems, state values are estimated using traditional least squares techniques based on redundant sets of measurements. Often these measurement models include additional parameters, whose errors affect the accuracy of the state estimates. Usually these parameters are known within certain accuracy limits based on manufacturing specifications or from prior calibration procedures. In complex measurement scenarios, however, these additional parameters may not be known as precisely as desired.

In this chapter, the "consider" methodology will be explored first from a batch estimation perspective using least squares, minimum variance, and maximum likelihood derivations. Different cases will be described based on the *a priori* information available. Next, the consider analysis will be expanded into a sequential, or recursive, formulation. Finally, two simple examples will be presented, one static and one dynamic. Focusing on the batch estimators, these examples will show the advantages and limitations of using the consider approach for state estimation.

A. Measurement Model Definition

A discrete linear measurement model with additional parameters is described as

$$\tilde{\boldsymbol{y}}_k = H_{x_k}\boldsymbol{x}_k + H_{p_k}\boldsymbol{p} + \boldsymbol{v}_k, \qquad \boldsymbol{v}_k \sim \mathrm{N}(0, R_k) \tag{2.1}$$

where $\tilde{\boldsymbol{y}}$ is a $m \times 1$ vector containing the collected measurements, $\boldsymbol{x}$ is a $n \times 1$ vector containing the states to be estimated, $\boldsymbol{p}$ is a $r \times 1$ vector containing the parameter values, H_x and H_p are the measurement sensitivity matrices, k is the measurement

step, and $\boldsymbol{v}$ is a $m \times 1$ vector containing the measurement noise. The nomenclature $\mathrm{N}(a, B)$ denotes a normal distribution with mean a and covariance B. Often parameters appear nonlinearly with the states, but following traditional linearization techniques Eq. (2.1) can still be used.

B. Batch Estimation

For batch estimation, a set of q measurements can be collected as

$$\tilde{\boldsymbol{y}} = \begin{bmatrix} \tilde{\boldsymbol{y}}_1 \\ \tilde{\boldsymbol{y}}_2 \\ \vdots \\ \tilde{\boldsymbol{y}}_q \end{bmatrix}, \qquad H_x = \begin{bmatrix} H_{x_1} \\ H_{x_2} \\ \vdots \\ H_{x_q} \end{bmatrix}, \qquad H_p = \begin{bmatrix} H_{p_1} \\ H_{p_2} \\ \vdots \\ H_{p_q} \end{bmatrix}, \qquad \boldsymbol{v} = \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \\ \vdots \\ \boldsymbol{v}_q \end{bmatrix} \tag{2.2}$$

$$E\left\{\boldsymbol{v}\boldsymbol{v}^{\mathrm{T}}\right\} = R = \begin{bmatrix} R_1 & & & \\ & R_2 & & \\ & & \ddots & \\ & & & R_q \end{bmatrix} \tag{2.3}$$

where $E\{\dots\}$ is the expectation function and the individual measurement covariances, R_k, are defined by

$$E\left\{\boldsymbol{v}_i\right\} = 0, \qquad E\left\{\boldsymbol{v}_i \boldsymbol{v}_j^{\mathrm{T}}\right\} = R_{ij}\delta_{ij} \tag{2.4}$$

where δ_{ij} is the Kronecker delta function and $R_k = R_{ii}$. Since the consolidated vectors represent q sets of m measurements, the $\tilde{\boldsymbol{y}}$ and $\boldsymbol{v}$ vectors are $qm \times 1$ vectors. Similarly, the H_x and H_p matrices are $qm \times n$ and $qm \times r$, respectively.

1. Least Squares

If the system is static (i.e., $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$), then one possible method to account for the parameters in the batch state estimate is to create an adjusted measurement vector

$$\bar{\boldsymbol{y}} = \tilde{\boldsymbol{y}} - H_p \boldsymbol{p} \tag{2.5}$$

The weighted least squares cost function is then given by

$$J = \frac{1}{2} \left(\bar{\boldsymbol{y}} - \hat{\boldsymbol{y}} \right)^{\mathrm{T}} W \left(\bar{\boldsymbol{y}} - \hat{\boldsymbol{y}} \right) \tag{2.6}$$

where W is a weighting matrix. The estimated measurements, $\hat{\boldsymbol{y}}$, are defined as

$$\hat{\boldsymbol{y}} = H_x \hat{\boldsymbol{x}} \tag{2.7}$$

where $\hat{\boldsymbol{x}}$ are the estimated states. Assuming matrices H_x, H_p, and vector $\boldsymbol{p}$ are given known constants, then minimizing Eq. (2.6) results in the traditional weighted least squares solution: [13]

$$\hat{\boldsymbol{x}} = \left(H_x^{\mathrm{T}} W H_x \right)^{-1} H_x^{\mathrm{T}} W \bar{\boldsymbol{y}} = \left(H_x^{\mathrm{T}} W H_x \right)^{-1} H_x^{\mathrm{T}} W \left(\tilde{\boldsymbol{y}} - H_p \boldsymbol{p} \right) \tag{2.8}$$

If the true values of the parameters are used and the weight matrix is defined as $W = R^{-1}$, the covariance of $\hat{\boldsymbol{x}}$ is the familiar result

$$P_{xx} = E \left\{ \left(\hat{\boldsymbol{x}} - \boldsymbol{x} \right) \left(\hat{\boldsymbol{x}} - \boldsymbol{x} \right)^{\mathrm{T}} \right\} = \left(H_x^{\mathrm{T}} R^{-1} H_x \right)^{-1} \tag{2.9}$$

If, however, the true values of the parameters are not known and only approximate values are used then the covariance of the states is no longer the result shown in Eq. (2.9). Additional calculations are needed to properly estimate the state covariance.

One alternative solution is to estimate the parameters using an augmented state vector

$$\boldsymbol{z} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{p} \end{bmatrix} \tag{2.10}$$

This redefines the true measurement model as

$$\tilde{\boldsymbol{y}} = H_z \boldsymbol{z} + \boldsymbol{v} \tag{2.11}$$

where $H_z = \begin{bmatrix} H_x & H_p \end{bmatrix}$. Replacing the new measurement equation for $\bar{\boldsymbol{y}}$ and letting $\hat{\boldsymbol{y}} = H_z \hat{\boldsymbol{z}}$ in Eq. (2.6) produces the augmented least squares solution

$$\hat{\boldsymbol{z}} = \left(H_z^{\mathrm{T}} W H_z\right)^{-1} H_z^{\mathrm{T}} W \tilde{\boldsymbol{y}} \tag{2.12}$$

The covariance for the augmented state vector is now easily proven to be

$$\begin{aligned} P_z &= E\left\{(\hat{\boldsymbol{z}} - \boldsymbol{z})(\hat{\boldsymbol{z}} - \boldsymbol{z})^{\mathrm{T}}\right\} \\ &= E\left\{ \begin{bmatrix} (\hat{\boldsymbol{x}} - \boldsymbol{x}) \\ (\hat{\boldsymbol{p}} - \boldsymbol{p}) \end{bmatrix} \begin{bmatrix} (\hat{\boldsymbol{x}} - \boldsymbol{x}) \\ (\hat{\boldsymbol{p}} - \boldsymbol{p}) \end{bmatrix}^{\mathrm{T}} \right\} = \begin{bmatrix} P_{xx} & P_{xp} \\ P_{px} & P_{pp} \end{bmatrix} = \left(H_z^{\mathrm{T}} W H_z\right)^{-1} \end{aligned} \tag{2.13}$$

where the submatrices are

$$P_{xx} = \left(H_x^{\mathrm{T}} W H_x - H_x^{\mathrm{T}} W H_p \left(H_p^{\mathrm{T}} W H_p\right)^{-1} H_p^{\mathrm{T}} W H_x\right)^{-1} \tag{2.14}$$

$$P_{xp} = -P_{xx} H_x^{\mathrm{T}} W H_p \left(H_p^{\mathrm{T}} W H_p\right)^{-1} \tag{2.15}$$

$$P_{pp} = \left(H_p^{\mathrm{T}} W H_p - H_p^{\mathrm{T}} W H_x \left(H_x^{\mathrm{T}} W H_x\right)^{-1} H_x^{\mathrm{T}} W H_p\right)^{-1} \tag{2.16}$$

$$P_{px} = -P_{pp} H_p^{\mathrm{T}} W H_x \left(H_x^{\mathrm{T}} W H_x\right)^{-1} = P_{xp}^{\mathrm{T}} \tag{2.17}$$

Note that this expansion separates out the covariances between just the states, P_{xx}, and just the parameters, P_{pp}. The additional matrix that results, P_{xp}, contains the covariance terms between the states and the parameters. In the remainder of the text this matrix will be known as the cross-covariance.

Solving Eq. (2.12) for only the state estimates, $\hat{\boldsymbol{x}}$, results in the consider least squares (CLS) estimate

$$\hat{\boldsymbol{x}} = \left(P_{xx} H_x^{\mathrm{T}} + P_{xp} H_p^{\mathrm{T}}\right) W \tilde{\boldsymbol{y}} \tag{2.18}$$

The result in Eq. (2.18) can be thought of as the weighted sum of the measurements with respect to the state and the parameter estimates where the weights are defined by the covariance matrices P_{xx} and P_{xp}. Notice that the correlation or cross-covariance, P_{xp}, of the state estimate error to the parameter estimate error must be "considered" in Eq. (2.18), whether or not we seek to use a companion equation to actually estimate the parameters, $\boldsymbol{p}$.

2. Minimum Variance with *A Priori* Estimates

Minimum variance techniques provide the optimal estimate of the states based on probability. Formally, a minimum variance estimator is designed such that every element $\hat{x}_i$ of $\hat{\boldsymbol{x}}$ is as close as possible to the true value x_i, as measured by the error variance $\sigma_{x_i}^2 \equiv E\left\{(\hat{x}_i - x_i)^2\right\}$. For linear systems, it can be proven that the minimum variance estimator also minimizes the trace of the estimation error covariance matrix, which is $\sum_{i=1}^{n} \sigma_{x_i}^2$ [13].

For consider analysis, the augmented state vector, $\boldsymbol{z}$, is used to find the optimal minimum variance solution and then the estimates for the states, $\hat{\boldsymbol{x}}$, are extracted from this solution. The first derivation will explore the case where *a priori* estimates of both the states and the parameters are available. Special cases will then be shown where only state or parameter estimates are available *a priori*, as well as, the special case when no initial estimates are available. Finally, the case where the parameters are known exactly will be investigated as a verification of previous developments. *A priori* estimates are estimates based on information known before a particular set of measurements are included. *A posteriori* estimates, on the other hand, are estimates which include all of the *a priori* information as well as the current set of measurements. In the remainder of the text, a $^-$ will be used to denote the *a priori* estimates while a $^+$ will be used to denote the *a posteriori* estimates.

The minimum variance derivation is based on the assumption that the best estimate of $\boldsymbol{x}$ can be found from a linear combination of the measurements and its *a priori* estimates

$$\hat{\boldsymbol{x}}^{+} = M\tilde{\boldsymbol{y}} + N\hat{\boldsymbol{x}}^{-} \tag{2.19}$$

where M and N are matrices to be determined. In addition to being linear, minimum variance estimators are also designed to be unbiased. An estimator is defined as unbiased if $E\{\hat{\boldsymbol{x}}\} = \boldsymbol{x}$ for all possible values of $\boldsymbol{x}$ given $\tilde{\boldsymbol{y}}$ [13].

In the case where there are additional parameters in the measurement model as in Eq. (2.1) and the true value of the parameters are known, then $\tilde{\boldsymbol{y}}$ can be replaced by $\bar{\boldsymbol{y}}$ as before. The solution is then known to be [13]

$$\begin{aligned}\hat{\boldsymbol{x}}^{+} &= \left(H_x^{\mathrm{T}} R^{-1} H_x + P_{xx}^{-^{-1}}\right)^{-1} \left(H_x^{\mathrm{T}} R^{-1} \bar{\boldsymbol{y}} + P_{xx}^{-^{-1}} \hat{\boldsymbol{x}}^{-}\right) \\ &= \left(H_x^{\mathrm{T}} R^{-1} H_x + P_{xx}^{-^{-1}}\right)^{-1} \left(H_x^{\mathrm{T}} R^{-1} \tilde{\boldsymbol{y}} + P_{xx}^{-^{-1}} \hat{\boldsymbol{x}}^{-} - H_x^{\mathrm{T}} R^{-1} H_p \boldsymbol{p}\right)\end{aligned} \tag{2.20}$$

Once again, however, the true values of the parameters are not typically known and only estimates are available. As a result, the augmented state vector, $\boldsymbol{z}$, is used instead to produce

$$\hat{\boldsymbol{z}}^{+} = \left(H_z^{\mathrm{T}} R^{-1} H_z + P_z^{-^{-1}}\right)^{-1} \left(H_z^{\mathrm{T}} R^{-1} \tilde{\boldsymbol{y}} + P_z^{-^{-1}} \hat{\boldsymbol{z}}^{-}\right) \tag{2.21}$$

where P_z^{-} is the covariance of the augmented *a priori* estimates, $\hat{\boldsymbol{z}}^{-}$. Additionally, the updated covariance, P_z^{+}, is known to be [13]

$$P_z^{+} = E\left\{\left(\hat{\boldsymbol{z}}^{+} - \boldsymbol{z}\right)\left(\hat{\boldsymbol{z}}^{+} - \boldsymbol{z}\right)^{\mathrm{T}}\right\} = \left(H_z^{\mathrm{T}} R^{-1} H_z + P_z^{-^{-1}}\right)^{-1} \tag{2.22}$$

Let $P_z^{-^{-1}}$ be defined by

$$P_z^{-^{-1}} \triangleq \begin{bmatrix} M_{xx} & M_{xp} \\ M_{px} & M_{pp} \end{bmatrix} \tag{2.23}$$

then

$$P_z^{-^{-1}} P_z^- = \begin{bmatrix} M_{xx} & M_{xp} \\ M_{px} & M_{pp} \end{bmatrix} \begin{bmatrix} P_{xx}^- & P_{xp}^- \\ P_{px}^- & P_{pp} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \tag{2.24}$$

Solving for the submatrices of $P_z^{-^{-1}}$ yields

$$M_{xx} = P_{xx}^{-^{-1}} + P_{xx}^{-^{-1}} P_{xp}^- \left(P_{pp} - P_{px}^- P_{xx}^{-^{-1}} P_{xp}^- \right)^{-1} P_{px}^- P_{xx}^{-^{-1}} = \left(P_{xx}^- - P_{xp}^- P_{pp}^{-1} P_{px}^- \right)^{-1} \tag{2.25}$$

$$M_{xp} = -\left(P_{xx}^- - P_{xp}^- P_{pp}^{-1} P_{px}^- \right)^{-1} P_{xp}^- P_{pp}^{-1} = -M_{xx} P_{xp}^- P_{pp}^{-1} \tag{2.26}$$

$$M_{px} = -\left(P_{pp} - P_{px}^- P_{xx}^{-^{-1}} P_{xp}^- \right)^{-1} P_{px}^- P_{xx}^{-^{-1}} = -M_{pp} P_{px}^- P_{xx}^{-^{-1}} = M_{xp}^{\mathrm{T}} \tag{2.27}$$

$$M_{pp} = P_{pp}^{-1} + P_{pp}^{-1} P_{px}^- \left(P_{xx}^- - P_{xp}^- P_{pp}^{-1} P_{px}^- \right)^{-1} P_{xp}^- P_{pp}^{-1} = \left(P_{pp} - P_{px}^- P_{xx}^{-^{-1}} P_{xp}^- \right)^{-1} \tag{2.28}$$

Expanding out the matrices in Eq. (2.21) gives

$$\begin{bmatrix} H_x^{\mathrm{T}} R^{-1} H_x + M_{xx} & H_x^{\mathrm{T}} R^{-1} H_p + M_{xp} \\ H_p^{\mathrm{T}} R^{-1} H_x + M_{px} & H_p^{\mathrm{T}} R^{-1} H_p + M_{pp} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}^+ \\ \hat{\boldsymbol{p}}^+ \end{bmatrix} = \begin{bmatrix} H_x^{\mathrm{T}} R^{-1} \tilde{\boldsymbol{y}} + M_{xx} \hat{\boldsymbol{x}}^- + M_{xp} \hat{\boldsymbol{p}}^- \\ H_p^{\mathrm{T}} R^{-1} \tilde{\boldsymbol{y}} + M_{px} \hat{\boldsymbol{x}}^- + M_{pp} \hat{\boldsymbol{p}}^- \end{bmatrix} \tag{2.29}$$

Tapley *et al.* [8] show that the solution for the state estimate is

$$\hat{\boldsymbol{x}}^+ = \left(P_{xx}^+ H_x^{\mathrm{T}} + P_{xp}^+ H_p^{\mathrm{T}} \right) R^{-1} \tilde{\boldsymbol{y}} + \left(P_{xx}^+ M_{xx} + P_{xp}^+ M_{px} \right) \hat{\boldsymbol{x}}^- + \left(P_{xx}^+ M_{xp} + P_{xp}^+ M_{pp} \right) \hat{\boldsymbol{p}} \tag{2.30}$$

where $\hat{\boldsymbol{p}} = \hat{\boldsymbol{p}}^-$ and

$$P_{xx}^+ = \Big(H_x^{\mathrm{T}} R^{-1} H_x + M_{xx} - \left(H_x^{\mathrm{T}} R^{-1} H_p + M_{xp} \right) \times \left(H_p^{\mathrm{T}} R^{-1} H_p + M_{pp} \right)^{-1} \left(H_p^{\mathrm{T}} R^{-1} H_x + M_{px} \right) \Big)^{-1} \tag{2.31}$$

$$P_{xp}^+ = -P_{xx}^+ \left(H_x^{\mathrm{T}} R^{-1} H_p + M_{xp} \right) \left(H_p^{\mathrm{T}} R^{-1} H_p + M_{pp} \right)^{-1} \tag{2.32}$$

which are the updated state and cross-covariances, respectively. This means that given q sets of m measurements, $\tilde{\boldsymbol{y}}$, *a priori* estimates of the states, $\hat{\boldsymbol{x}}^-$, and the

parameters, $\hat{\boldsymbol{p}}$, and their associated covariance, P_z^-, it is possible to find an updated value of the states from Eq. (2.30).

There are instances, however, where all of this information is not available *a priori* and an alternate solution must be formed. The next few sections investigate these alternatives.

a. Special Case 1: No Parameter Estimates *A Priori*

The analysis presented in the previous section relied on the assumption that *a priori* estimates of both the states and parameters are available. Depending on the desired implementation, either only state estimates or parameter estimates may be available. Consider first the case where only state estimates are known *a priori*. In other words, there is no knowledge on the possible values of the measurement model parameters. Mathematically, this means $P_{pp} \to \infty$ or $P_{pp}^{-1} \to 0$ which forces $M_{xp} = M_{px} = M_{pp} = 0$ from Eqs. (2.26), (2.27), and (2.28), respectively. Additionally, M_{xx} reduces to $P_{xx}^{-^{-1}}$ from Eq. (2.25). Thus, Eq. (2.30) becomes

$$\hat{\boldsymbol{x}}^+ = \left(P_{xx}^+ H_x^{\mathrm{T}} + P_{xp}^+ H_p^{\mathrm{T}}\right) R^{-1} \tilde{\boldsymbol{y}} + \left(P_{xx}^+ P_{xx}^{-^{-1}}\right) \hat{\boldsymbol{x}}^- \tag{2.33}$$

and the updated covariances are now

$$P_{xx}^+ = \left(P_{xx}^{-^{-1}} + H_x^{\mathrm{T}} R^{-1} H_x - H_x^{\mathrm{T}} R^{-1} H_p \left(H_p^{\mathrm{T}} R^{-1} H_p\right)^{-1} H_p^{\mathrm{T}} R^{-1} H_x\right)^{-1} \tag{2.34}$$

$$P_{xp}^+ = -P_{xx}^+ H_x^{\mathrm{T}} R^{-1} H_p \left(H_p^{\mathrm{T}} R^{-1} H_p\right)^{-1} \tag{2.35}$$

Recollecting terms in Eq. (2.33) also shows striking similarities to Eq. (2.20), only now there are additional terms to help account for the parameter uncertainty.

b. Special Case 2: No State Estimates *A Priori*

As an alternative to the developments above, consider the case where there are no *a priori* estimates for the states but knowledge is available for the measurement model parameters. In this scenario, $P_{xx} \to \infty$ or $P_{xx}^{-1} \to 0$ which forces $M_{xx} = M_{xp} = M_{px} = 0$ and $M_{pp} = P_{pp}^{-1}$ from Eqs. (2.25) - (2.28). Substituting these values into Eq. (2.30) results in

$$\hat{\boldsymbol{x}}^{+} = \left(P_{xx}^{+} H_x^{\mathrm{T}} + P_{xp}^{+} H_p^{\mathrm{T}}\right) R^{-1} \tilde{\boldsymbol{y}} + \left(P_{xp}^{+} P_{pp}^{-1}\right) \hat{\boldsymbol{p}} \tag{2.36}$$

where the updated covariances equate to

$$P_{xx}^{+} = \left(H_x^{\mathrm{T}} R^{-1} H_x - H_x^{\mathrm{T}} R^{-1} H_p \left(H_p^{\mathrm{T}} R^{-1} H_p + P_{pp}^{-1}\right)^{-1} H_p^{\mathrm{T}} R^{-1} H_x\right)^{-1} \tag{2.37}$$

$$P_{xp}^{+} = -P_{xx}^{+} H_x^{\mathrm{T}} R^{-1} H_p \left(H_p^{\mathrm{T}} R^{-1} H_p + P_{pp}^{-1}\right)^{-1} \tag{2.38}$$

By invoking the Sherman-Morrison-Woodbury Matrix Inversion Lemma (described in Appendix B) with

$$A = P_{pp}^{-1}, \qquad B = H_p^{\mathrm{T}}, \qquad C = R^{-1}, \qquad D = H_p \tag{2.39}$$

the coefficient in front of $\hat{\boldsymbol{p}}$ in Eq. (2.36) reduces to

$$P_{xx}^{+} H_x^{\mathrm{T}} R^{-1} H_p \left(H_p^{\mathrm{T}} R^{-1} H_p + P_{pp}^{-1}\right)^{-1} P_{pp}^{-1} = P_{xx}^{+} H_x^{\mathrm{T}} \left(H_p P_{pp} H_p^{\mathrm{T}} + R\right)^{-1} H_p \tag{2.40}$$

and the solution for the updated state estimate becomes

$$\hat{\boldsymbol{x}}^{+} = \left(P_{xx}^{+} H_x^{\mathrm{T}} + P_{xp}^{+} H_p^{\mathrm{T}}\right) R^{-1} \tilde{\boldsymbol{y}} - P_{xx}^{+} H_x^{\mathrm{T}} \left(H_p P_{pp} H_p^{\mathrm{T}} + R\right)^{-1} H_p \hat{\boldsymbol{p}} \tag{2.41}$$

c. Special Case 3: No *A Priori* Estimates

In the case where no *a priori* information is available for either the states or the parameters then $P_{xx} \to \infty$ or $P_{xx}^{-1} \to 0$ and $P_{pp} \to \infty$ or $P_{pp}^{-1} \to 0$. By substituting

in the former to Eq. (2.33) and the latter into Eq. (2.41) both equations reduce to

$$\hat{\boldsymbol{x}}^{+} = \left(P_{xx}^{+} H_x^{\mathrm{T}} + P_{xp}^{+} H_p^{\mathrm{T}}\right) R^{-1} \tilde{\boldsymbol{y}} \tag{2.42}$$

where

$$P_{xx}^{+} = \left(H_x^{\mathrm{T}} R^{-1} H_x - H_x^{\mathrm{T}} R^{-1} H_p \left(H_p^{\mathrm{T}} R^{-1} H_p\right)^{-1} H_p^{\mathrm{T}} R^{-1} H_x\right)^{-1} \tag{2.43}$$

$$P_{xp}^{+} = -P_{xx}^{+} H_x^{\mathrm{T}} R^{-1} H_p \left(H_p^{\mathrm{T}} R^{-1} H_p\right)^{-1} \tag{2.44}$$

under these initial conditions. Comparing Eqs. (2.42) - (2.44) to Eqs. (2.14), (2.15), and (2.18) makes it apparent that the batch consider least squares solution found previously is also the minimum variance consider least squares result if $W = R^{-1}$. This implies that the optimal estimate of the states given no prior knowledge from both a least squares and a minimum variance perspective is given by Eqs. (2.42) - (2.44).

d. Special Case 4: True Parameter Values Known

If the true parameter values are known then $\hat{\boldsymbol{p}} - \boldsymbol{p} = 0$ and $P_{pp} = P_{xp} = P_{px} = 0$. Substituting this into Eqs. (2.37) and (2.38) reduces P_{xx}^{+} and P_{xp}^{+} to

$$P_{xx}^{+} = \left(H_x^{\mathrm{T}} R^{-1} H_x\right)^{-1}, \qquad P_{xp}^{+} = 0 \tag{2.45}$$

since $\left(H_p^{\mathrm{T}} R^{-1} H_p + P_{pp}^{-1}\right)^{-1} \to 0$. Eq. (2.41) is now

$$\hat{\boldsymbol{x}} = \left(H_x^{\mathrm{T}} R^{-1} H_x\right)^{-1} H_x^{\mathrm{T}} R^{-1} \left(\tilde{\boldsymbol{y}} - H_p \boldsymbol{p}\right) \tag{2.46}$$

which is just the traditional least squares (minimum variance) solution with $W = R^{-1}$ as shown in Eq. (2.8). Furthermore, under this condition $M_{xx} = P_{xx}^{-^{-1}}$ and Eq. (2.30) becomes

$$\hat{\boldsymbol{x}}^{+} = \left(H_x^{\mathrm{T}} R^{-1} H_x + P_{xx}^{-^{-1}}\right)^{-1} \left(H_x^{\mathrm{T}} R^{-1} \tilde{\boldsymbol{y}} + P_{xx}^{-^{-1}} \hat{\boldsymbol{x}}^{-} - H_x^{\mathrm{T}} R^{-1} H_p \boldsymbol{p}\right) \tag{2.47}$$

which is equivalent to Eq. (2.20).

3. Maximum Likelihood

An alternate measure of probability-based optimality is through the use of maximum likelihood. Given a set of measurements, $\tilde{\boldsymbol{y}}$, maximum likelihood finds the optimal states, $\boldsymbol{x}$, that maximize the likelihood function, L. In mathematical terms this can be expressed as

$$L\left(\tilde{\boldsymbol{y}}|\boldsymbol{x}\right) = \prod_{i=1}^{p} f_i\left(\tilde{\boldsymbol{y}}|\boldsymbol{x}\right) \tag{2.48}$$

where f represents a probability density function (pdf), p is the total number of probability density functions, and the expression $\tilde{\boldsymbol{y}}|\boldsymbol{x}$ represents the conditional probability of $\tilde{\boldsymbol{y}}$ given $\boldsymbol{x}$ [13].

Given the augmented measurement model defined in Eq. (2.11), the probability density function of the measurement vector, $\tilde{\boldsymbol{y}}$, can be defined as

$$f\left(\tilde{\boldsymbol{y}}|\boldsymbol{z}\right) = \frac{1}{\left(2\pi\right)^{qm/2}\left|R\right|^{1/2}} \exp\left\{-\frac{1}{2}\left[\tilde{\boldsymbol{y}} - H_z\boldsymbol{z}\right]^{\mathrm{T}} R^{-1}\left[\tilde{\boldsymbol{y}} - H_z\boldsymbol{z}\right]\right\} \tag{2.49}$$

where $|R|$ denotes the determinant of the matrix R. The first and second central moments of $\tilde{\boldsymbol{y}}$ are

$$E\left\{\tilde{\boldsymbol{y}}\right\} = \boldsymbol{\mu}_y = H_z\boldsymbol{z} \tag{2.50}$$

$$\mathrm{cov}\left\{\tilde{\boldsymbol{y}}\right\} = E\left\{\left[\tilde{\boldsymbol{y}} - \boldsymbol{\mu}_y\right]\left[\tilde{\boldsymbol{y}} - \boldsymbol{\mu}_y\right]^{\mathrm{T}}\right\} = R \tag{2.51}$$

Many likelihood functions include monotonic exponential terms due to Gaussian or approximately Gaussian distributions. Because of this, the natural logarithm can be applied to the likelihood function since it also is a monotonic function. Taking the natural logarithm of Eq. (2.49) gives

$$\ln\left[L\left(\tilde{\boldsymbol{y}}|\boldsymbol{z}\right)\right] = -\frac{1}{2}\left[\tilde{\boldsymbol{y}} - H_z\boldsymbol{z}\right]^{\mathrm{T}} R^{-1}\left[\tilde{\boldsymbol{y}} - H_z\boldsymbol{z}\right] - \frac{m}{2}\ln\left(2\pi\right) - \frac{1}{2}\ln\left(\left|R\right|\right) \tag{2.52}$$

which is also known as the log-likelihood function. Since the last two terms are constant with respect to the states, they can be neglected. Also, taking the negative

of a maximizing function makes it a minimizing function or

$$J(\boldsymbol{z}) = -\ln\left[L\left(\tilde{\boldsymbol{y}}|\boldsymbol{z}\right)\right] = \frac{1}{2}\left[\tilde{\boldsymbol{y}} - H_z\boldsymbol{z}\right]^{\mathrm{T}} R^{-1}\left[\tilde{\boldsymbol{y}} - H_z\boldsymbol{z}\right] \tag{2.53}$$

which is the same as the augmented least squares cost function discussed in Section 1 with $W = R^{-1}$. While the above discussion is simply a rederivation of the classic maximum likelihood derivation for batch least squares, this analysis verifies that the optimal state estimate provided by the consider methodology is just the state estimate component of the solution to the augmented cost function.

4. Cramer-Rao Bound

One of the major benefits of using a maximum likelihood approach is that it enables us to take advantage of one of the most powerful tools in estimation theory, the Cramer-Rao inequality. The power of the Cramer-Rao inequality is that it gives a lower bound on the expected errors between the estimated quantities and the true values based on the known statistical properties of the measurement errors. The Cramer-Rao inequality for an unbiased estimate $\hat{\boldsymbol{x}}$ is given by

$$P \equiv E\left\{\left(\hat{\boldsymbol{x}} - \boldsymbol{x}\right)\left(\hat{\boldsymbol{x}} - \boldsymbol{x}\right)^{\mathrm{T}}\right\} \geq F^{-1} \tag{2.54}$$

where the Fisher information matrix, F, is given by

$$F = E\left\{\left[\frac{\partial}{\partial \boldsymbol{x}}\ln f\left(\tilde{\boldsymbol{y}}|\boldsymbol{x}\right)\right]\left[\frac{\partial}{\partial \boldsymbol{x}}\ln f\left(\tilde{\boldsymbol{y}}|\boldsymbol{x}\right)\right]^{\mathrm{T}}\right\} = -E\left\{\frac{\partial^2}{\partial \boldsymbol{x}\partial \boldsymbol{x}^{\mathrm{T}}}\ln f\left(\tilde{\boldsymbol{y}}|\boldsymbol{x}\right)\right\} \tag{2.55}$$

The first- and second-order derivatives are assumed to exist and to be absolutely integrable [13].

Using the maximum likelihood cost function defined in Eq. (2.53) the Cramer-Rao lower bound is known to be [13]

$$P_z \geq \left(H_z^{\mathrm{T}} R^{-1} H_z\right)^{-1} \tag{2.56}$$

Comparing this result to the covariance definitions found for the least squares solution, Eq. (2.13), proves that when the weight matrix, W, is equal to the inverse of the measurement covariance, R^{-1}, the augmented state covariance is the optimal covariance according to both least squares, minimum variance, and maximum likelihood measures of optimality. More importantly, the state covariance, P_{xx}, provided by batch consider methods is also optimal since it is simply the submatrix associated with only the states from the augmented state covariance.

5. Bayesian Estimation

Maximum likelihood is a great tool, but for the least squares problem it does not account for *a priori* knowledge of the states and/or parameters. Bayesian estimation provides a way to combine the *a priori* augmented state information with the measurements through a conditional density function, $f\left(\boldsymbol{x}|\tilde{\boldsymbol{y}}\right)$. The conditional density function is defined through Bayes' rule

$$f\left(\boldsymbol{x}|\tilde{\boldsymbol{y}}\right) = \frac{f\left(\tilde{\boldsymbol{y}}|\boldsymbol{x}\right) f\left(\boldsymbol{x}\right)}{f\left(\tilde{\boldsymbol{y}}\right)} \tag{2.57}$$

Since the purpose of Bayesian estimation is to find the optimal *a posteriori* estimate of the states, the probability density function of the measurements, $f\left(\tilde{\boldsymbol{y}}\right)$, is a constant and can be neglected. Using the modified Bayes' equation and taking its maximum results in maximum *a posteriori* (MAP) estimation. Similar to maximum likelihood, taking the negative of the natural logarithm of the Bayes' equation results in the cost function

$$J_{\text{MAP}}\left(\hat{\boldsymbol{z}}^{+}\right) = \ln\left[f\left(\tilde{\boldsymbol{y}}|\hat{\boldsymbol{z}}^{+}\right)\right] + \ln\left[f\left(\hat{\boldsymbol{z}}^{+}\right)\right] \tag{2.58}$$

Notice that the first term is the natural logarithm of the likelihood function and the second term depends on the *a priori* information of the states being estimated. Thus, if the *a priori* information for $f\left(\hat{\boldsymbol{z}}^{+}\right)$ has a uniform distribution, then the MAP

solution is also the maximum likelihood solution. Since $f\left(\tilde{\boldsymbol{y}}|\hat{\boldsymbol{z}}^{+}\right)$ is already given by Eq. (2.49), only $f\left(\hat{\boldsymbol{z}}^{+}\right)$ needs to be determined. Given the *a priori* state estimates, $\hat{\boldsymbol{z}}^{-}$, their mean, $\hat{\boldsymbol{z}}^{+}$, and covariance, P_0, their pdf is given by

$$f\left(\hat{\boldsymbol{z}}^{+}\right)=\frac{1}{(2\pi)^{(n+r)/2}\left|P_0\right|^{1/2}}\exp\left\{-\frac{1}{2}\left[\hat{\boldsymbol{z}}^{+}-\hat{\boldsymbol{z}}^{-}\right]^{\mathrm{T}}P_0^{-1}\left[\hat{\boldsymbol{z}}^{+}-\hat{\boldsymbol{z}}^{-}\right]\right\} \tag{2.59}$$

Using Eqs. (2.49) and (2.59) and maximizing Eq. (2.58) yields

$$\hat{\boldsymbol{z}}^{+}=\left(H_z^{\mathrm{T}}R^{-1}H_z+P_0^{-1}\right)^{-1}\left(H_z^{\mathrm{T}}R^{-1}\tilde{\boldsymbol{y}}+P_0^{-1}\hat{\boldsymbol{z}}^{-}\right) \tag{2.60}$$

which is equivalent to results produced using minimum variance methods with *a priori* state and parameter information.

a. Cramer-Rao Bound for Bayesian Estimation

The Cramer-Rao inequality can be extended for use with Bayesian estimation and is given by [13]

$$P\equiv E\left\{(\hat{\boldsymbol{x}}-\boldsymbol{x})(\hat{\boldsymbol{x}}-\boldsymbol{x})^{\mathrm{T}}\right\}\geq\left[F+E\left\{\left[\frac{\partial}{\partial\boldsymbol{x}}\ln f(\boldsymbol{x})\right]\left[\frac{\partial}{\partial\boldsymbol{x}}\ln f(\boldsymbol{x})\right]^{\mathrm{T}}\right\}\right]^{-1} \tag{2.61}$$

The Fisher information matrix, F, is already defined by $H_z^{\mathrm{T}}R^{-1}H_z$, thus, only the second term needs to be determined. Using the *a priori* probability density function gives

$$\begin{aligned}E\left\{\left[\frac{\partial}{\partial\hat{\boldsymbol{z}}^{+}}\ln f\left(\hat{\boldsymbol{z}}^{+}\right)\right]\left[\frac{\partial}{\partial\hat{\boldsymbol{z}}^{+}}\ln f\left(\hat{\boldsymbol{z}}^{+}\right)\right]^{\mathrm{T}}\right\}&=P_0^{-1}E\left\{\left(\hat{\boldsymbol{z}}^{+}-\hat{\boldsymbol{z}}^{-}\right)\left(\hat{\boldsymbol{z}}^{+}-\hat{\boldsymbol{z}}^{-}\right)^{\mathrm{T}}\right\}P_0^{-1}\\&=P_0^{-1}\end{aligned} \tag{2.62}$$

This gives a Cramer-Rao inequality lower bound of

$$P_z\geq\left[H_z^{\mathrm{T}}R^{-1}H_z+P_0^{-1}\right]^{-1} \tag{2.63}$$

but Eq. (2.22) verifies that this is in fact the minimum variance *a posteriori* covariance. Thus, once again, the updated covariance is also the optimal covariance verified by multiple measures of optimality for both fully augmented and consider methods of batch estimation.

C. Sequential Estimation

Although the batch results given in Section B provide theoretical insights into the nature of the consider least squares framework, batch estimators are often unwieldy for real world computations because of either very large data sets or limited computational power. It would be convenient to develop a sequential estimator based on the previously developed batch estimators.

1. Least Squares

Beginning again with the augmented state vector, $\boldsymbol{z}$, the sequential or recursive least squares solution is known to be [13]

$$\hat{\boldsymbol{z}}^{+} = \hat{\boldsymbol{z}}^{-} + K_z \left(\tilde{\boldsymbol{y}} - H_z \hat{\boldsymbol{z}}^{-} \right) \tag{2.64}$$

where

$$K_z = P_z^{-} H_z^{\mathrm{T}} \left(H_z P_z^{-} H_z^{\mathrm{T}} + W^{-1} \right)^{-1} \tag{2.65}$$

$$P_z^{+} = \left(I - K_z H_z \right) P_z^{-} \tag{2.66}$$

While the *a posteriori* and *a priori* notation used is more natural for dynamic systems where state estimates are updated once new measurements become available, for static systems a $^{-}$ can be thought of as the estimate at the k^{th} step and a $^{+}$ is the estimate at the $k+1$ step.

By multiplying out the block matrices of Eqs. (2.64) and (2.65), the updated state estimate is shown to be

$$\hat{\boldsymbol{x}}^{+} = \hat{\boldsymbol{x}}^{-} + K\left(\tilde{\boldsymbol{y}} - H_x\hat{\boldsymbol{x}}^{-} - H_p\hat{\boldsymbol{p}}\right) \tag{2.67}$$

where

$$K = \left(P_{xx}^{-}H_x^{\mathrm{T}} + P_{xp}^{-}H_p^{\mathrm{T}}\right)\left(H_xP_{xx}^{-}H_x^{\mathrm{T}} + H_xP_{xp}^{-}H_p^{\mathrm{T}} + H_pP_{px}^{-}H_x^{\mathrm{T}} + H_pP_{pp}H_p^{\mathrm{T}} + W^{-1}\right)^{-1} \tag{2.68}$$

$$P_{xx}^{+} = \left(I - KH_x\right)P_{xx}^{-} - KH_pP_{px}^{-} \tag{2.69}$$

$$P_{xp}^{+} = \left(I - KH_x\right)P_{xp}^{-} - KH_pP_{pp} \tag{2.70}$$

The augmented gain can also be written as [13]

$$K_z = P_z^{+}H_z^{\mathrm{T}}W \tag{2.71}$$

Based upon this representation of the augmented gain, the state gain is

$$K = \left(P_{xx}^{+}H_x^{\mathrm{T}} + P_{xp}^{+}H_p^{\mathrm{T}}\right)W \tag{2.72}$$

Note that this value of the gain is equivalent to using Eq. (2.18) as a differential update.

2. Minimum Variance

The minimum variance sequential estimator is found by beginning with Eq. (2.21) and using the Sherman-Morrison-Woodbury matrix inversion lemma with

$$A = P_z^{-^{-1}}, \qquad B = H_z^{\mathrm{T}}, \qquad C = R^{-1}, \qquad D = H_z \tag{2.73}$$

to produce

$$\hat{\boldsymbol{z}}^{+} = \left(P_z^{-} - P_z^{-}H_z^{\mathrm{T}}\left(H_zP_z^{-}H_z^{\mathrm{T}} + R\right)^{-1}H_zP_z^{-}\right)\left(H_z^{\mathrm{T}}R^{-1}\tilde{\boldsymbol{y}} + P_z^{-^{-1}}\hat{\boldsymbol{z}}^{-}\right) \tag{2.74}$$

Simplifying yields

$$\hat{\boldsymbol{z}}^+ = \hat{\boldsymbol{z}}^- + P_z^- H_z^{\mathrm{T}} \left(H_z P_z^- H_z^{\mathrm{T}} + R \right)^{-1} \left(\tilde{\boldsymbol{y}} - H_z \hat{\boldsymbol{z}}^- \right) \tag{2.75}$$

Expanding and solving for $\hat{\boldsymbol{x}}^+$ finds the minimum variance result

$$\hat{\boldsymbol{x}}^+ = \hat{\boldsymbol{x}}^- + K \left(\tilde{\boldsymbol{y}} - H_x \hat{\boldsymbol{x}}^- - H_p \hat{\boldsymbol{p}} \right) \tag{2.76}$$

where

$$K = \left(P_{xx}^- H_x^{\mathrm{T}} + P_{xp}^- H_p^{\mathrm{T}} \right) \left(H_x P_{xx}^- H_x^{\mathrm{T}} + H_x P_{xp}^- H_p^{\mathrm{T}} + H_p P_{px}^- H_x^{\mathrm{T}} + H_p P_{pp} H_p^{\mathrm{T}} + R \right)^{-1} \tag{2.77}$$

A verification of Eq. (2.77) can be achieved by minimizing the trace of the update state covariance, P_{xx}^+, with respect to the gain. Using Eq. (2.76) to evaluate the updated covariance generates

$$\begin{aligned} P_{xx}^+ &= E\left\{ \left(\hat{\boldsymbol{x}}^+ - \boldsymbol{x} \right) \left(\hat{\boldsymbol{x}}^+ - \boldsymbol{x} \right)^{\mathrm{T}} \right\} \\ &= \left(I - K H_x \right) P_{xx}^- \left(I - K H_x \right)^{\mathrm{T}} - \left(I - K H_x \right) P_{xp}^- H_p^{\mathrm{T}} K^{\mathrm{T}} - K H_p P_{px}^- \left(I - K H_x \right)^{\mathrm{T}} \\ &\quad + K H_p P_{pp} H_p^{\mathrm{T}} K^{\mathrm{T}} + K R K^{\mathrm{T}} \end{aligned} \tag{2.78}$$

which is the quadratic form or *Joseph's form* of the update state covariance [26]. Rearranging terms gives

$$\begin{aligned} P_{xx}^+ = P_{xx}^- &- K \left(H_x P_{xx}^- + H_p P_{px}^- \right) - \left(P_{xx}^- H_x^{\mathrm{T}} + P_{xp}^- H_p^{\mathrm{T}} \right) K^{\mathrm{T}} \\ &+ K \left(H_x P_{xx}^- H_x^{\mathrm{T}} + H_x P_{xp}^- H_p^{\mathrm{T}} + H_p P_{px}^- H_x^{\mathrm{T}} + H_p P_{pp} H_p^{\mathrm{T}} + R \right) K^{\mathrm{T}} \end{aligned} \tag{2.79}$$

The optimal gain is found by minimizing the cost function

$$J = \mathrm{tr}\left(P_{xx}^+ \right) \tag{2.80}$$

where $\mathrm{tr}\left(A \right)$ denotes the trace operation on the matrix A which sums the diagonal elements of A. Since in this particular case, the diagonal elements of P_{xx}^+ represent the

variances for each state, by minimizing the trace of the *a posteriori* state covariance, the minimum variance estimator is found. Taking the partial derivative of the above cost function with respect to the gain, K, and using the trace derivative properties found in Appendix A results in

$$\begin{aligned}\frac{\partial J}{\partial K} = 0 = 2K\left(H_x P_{xx}^- H_x^{\mathrm{T}} + H_x P_{xp}^- H_p^{\mathrm{T}} + H_p P_{px}^- H_x^{\mathrm{T}} + H_p P_{pp} H_p^{\mathrm{T}} + R\right) \\ - 2\left(P_{xx}^- H_x^{\mathrm{T}} + P_{xp}^- H_p^{\mathrm{T}}\right)\end{aligned} \tag{2.81}$$

Solving for the gain yields

$$K = \left(P_{xx}^- H_x^{\mathrm{T}} + P_{xp}^- H_p^{\mathrm{T}}\right)\left(H_x P_{xx}^- H_x^{\mathrm{T}} + H_x P_{xp}^- H_p^{\mathrm{T}} + H_p P_{px}^- H_x^{\mathrm{T}} + H_p P_{pp} H_p^{\mathrm{T}} + R\right)^{-1} \tag{2.82}$$

which is equivalent to the result shown in Eq. (2.77).

After comparing Eqs. (2.76) and (2.77) to Eqs. (2.67) and (2.68), it is apparent that the minimum variance sequential solution is equivalent to the sequential consider least squares result if $W = R^{-1}$ as before. Additionally, Eqs. (2.69) and (2.70) are still used to update the state and cross-covariances, respectively. Because the maximum likelihood and Bayesian batch estimators are equivalent to those developed using least squares and minimum variance, additional sequential estimators based on their optimality criterion do not need to be derived.

It is also possible to update the parameter estimates using the complete augmented state vector, however the consider least squares approach assumes that the parameter estimates, $\hat{\boldsymbol{p}}$, and their associated covariance, P_{pp}, do not change from their initial *a priori* estimates. Using this assumption in a batch estimator does not affect the optimality of the state estimate for consider analysis, but does affect the optimality for sequential estimators. The update for each measurement step is in itself optimal, however, since the parameter covariance is not being updated at the same time, the consider state estimate degrades from the augmented solution. Al-

though not described here, this can be observed theoretically by comparing the first few update steps for both the augmented and consider sequential estimators.

D. Static Estimation Examples

The theoretical results described above will be demonstrated through the use of two examples, one where the system is static, and the other will have known dynamics that can be described discretely via

$$\boldsymbol{x}_{k+1} = \Phi_k \boldsymbol{x}_k \tag{2.83}$$

where k is the step number and Φ is the state transition matrix with $\Phi_k = \Phi\left(k+1, k\right)$.

1. Example II.1

The static system example demonstrates the capabilities and limitations of the methods developed in Sections B and C. It is assumed that two sensors are available to provide measurement data of a stationary object. One sensor has an associated bias which can be represented by

$$\tilde{y}_1 = x + p + v_1 \qquad v_1 \sim \mathrm{N}(0, R_1) \tag{2.84}$$

The second sensor can estimate the parameter value directly or

$$\tilde{y}_2 = p + v_2 \qquad v_2 \sim \mathrm{N}(0, R_2) \tag{2.85}$$

It is assumed, though, that this measurement has a much higher variance than the first measurement ($R_2 \gg R_1$). This additional measurement makes the augmented system completely observable. If the augmented state vector was not fully observable then singularities would occur in Eq. (2.42). Observability is the measure of how well each state can be "observed" given the measurements available. A mathematical

test for observability is to ensure that the observability matrix has full rank and is described in most state-space control textbooks [1, 8, 13, 27]. Table 1 contains the values for the initial true and estimated values used in the analysis and following discussion.

Table 1: Values Used in Example II.1

Variable	x_0	$\hat{x}_0$	p	$\hat{p}$	P_{xx_0}	P_{xp_0}	P_{pp}	R_1	R_2
Value	4	4.2	0.4	0.3	0.04	0	0.01	0.01	100

Table 2: Results of Example II.1 - Single Run

Estimation Method	**Equation**	**Updated State** ($\hat{\boldsymbol{x}}^+$)	**Updated Parameter** ($\hat{\boldsymbol{p}}^+$)
Traditional Least Squares (TLS)	(2.8)	3.8962	N/A
TLS: State Estimates	(2.20)	3.8962	N/A
Consider Least Squares (CLS)	(2.42)	4.1823	N/A
CLS: State Estimates	(2.33)	4.1949	N/A
CLS: Parameter Estimates	(2.41)	3.9222	N/A
CLS: State & Parameter Estimates	(2.30)	3.9737	N/A
Augmented Least Squares (ALS)	(2.12)	4.1823	0.2139
ALS: State Estimates		4.1949	0.2012
ALS: Parameter Estimates		3.9222	0.4740
ALS: State & Parameter Estimates	(2.21)	3.9737	0.4226
Consider Sequential Least Squares	(2.76)	3.9284	N/A

Table 2 contains the results of a single test using the same 1000 sets of measurements for each of the methods described in the previous sections. Equation numbers

are provided to help correlate the method used and its theoretical development. The results of the fully augmented system have also been included for comparison purposes.

The first observation is that the traditional least squares (TLS) solution is off by almost the exact amount of the parameter error as a result of the linearity of the parameters and the assumption that they are perfectly known. When the other two estimators are given only state estimates *a priori*, the answer tends to be closer to those given values than the parameter estimates and vice versa, which is expected.

Comparing the consider least squares (CLS) results to the augmented least squares results (ALS) shows that they are in fact equivalent. This is also not surprising since the CLS results are simply the state estimates extracted from the ALS solution. It should be noted, however, that this is not always true, but a result of the linear examples being used in this discussion. The critical piece, however, is that in order to obtain these results, ALS had to invert a 2×2 matrix, while CLS only had to invert a 1×1 matrix. In this simple example the difference is trivial, but in larger systems considering the parameters can result in significant computational savings because a smaller state sensitivity matrix is inverted. Although not as significant, CLS also uses fewer multiplication and addition operations than the ALS by using only the updated state components of the ALS calculations.

While covariances of equal magnitudes were used in this example, as one covariance estimate increases with respect to the other, the state estimate will approach one of the aforementioned special cases. For example, if the initial covariance for the parameters was assumed to be 100 instead of 0.01, then an approximate solution for CLS with both initial estimates could be found using CLS with only initial state estimates. Similarly, if both initial covariances increase with respect to the measurement covariance then the solution to CLS with no initial estimates will be approached.

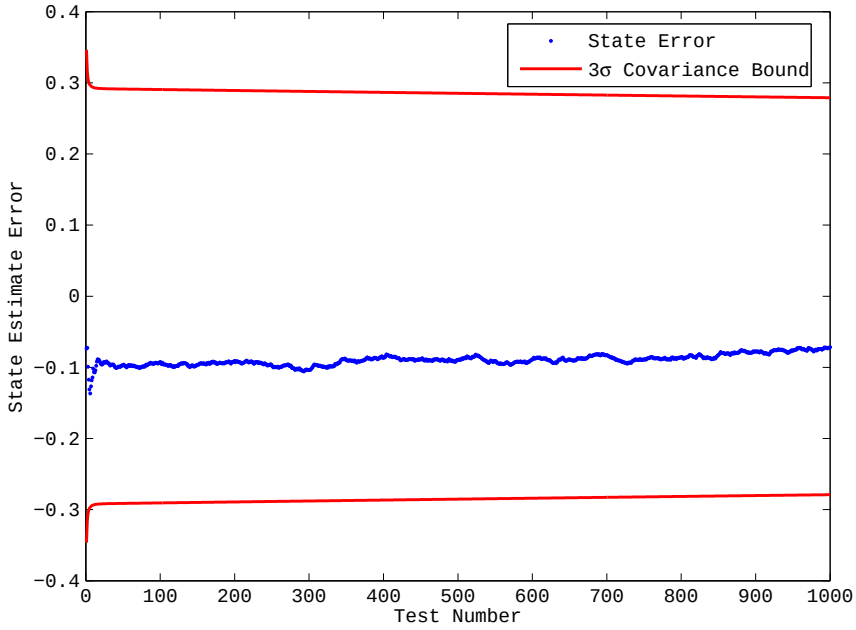


Fig. 1: Sequential Estimator Results for Example II.1

The individual measurement results for the sequential estimator are shown in Figure 1. Since the consider sequential estimator is non-optimal, these results are different from those produced by the consider and augmented batch estimators. It is important to note that the state covariance is converging and the state estimates stay comfortably within the covariance bounds. Other experiments show that if the initial covariance estimates were larger than those reported in this example, the estimator will still eventually converge, but requires more measurements.

To compare the updated state covariance value for each method, a Monte Carlo analysis was performed using 1000 simulations with 1000 measurements from each sensor per simulation. Figures 2 - 7 display the results. Each figure contains a histogram of the results combined with the estimated 3σ covariance bounds from each method.

While all the methods produced an approximate Gaussian distribution for the results, there are some discrepancies as to how the data points fall within the covariance bounds. Not surprisingly, there is a clear bias caused by the parameter error in both TLS methods that produces a distribution outside of the covariance bounds. There is also a bias, however, in all three cases where *a priori* estimates were provided to the CLS framework. This is due to the fact that in all cases a single value was given to the initial state and parameter estimates. Experiments show that if this value is allowed to vary, sampled from its assumed Gaussian distribution (where the true value was its mean and the covariance was given by their respective initial values), this bias disappears as the sample size grows. Thus, error in the initial estimates produces a bias in the resulting estimate during batch estimation. Additionally, the initial covariance estimates must be large enough to anticipate this error otherwise a large enough bias will place the mean estimate outside of the covariance bounds [28]. The CLS result with no initial estimates shows no bias from the true values, but at the cost of a much larger updated covariance than any of the other methods.

2. Example II.2

A linear oscillator is used to examine how the minimum variance batch estimators presented above can be used on dynamic systems. The dynamic equation for the undamped oscillator is given by

$$\ddot{x} + \omega_n^2 x = 0 \tag{2.86}$$

where ω_n is the natural frequency of the system. In state-space form these equations are written as

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = A\boldsymbol{x} \tag{2.87}$$

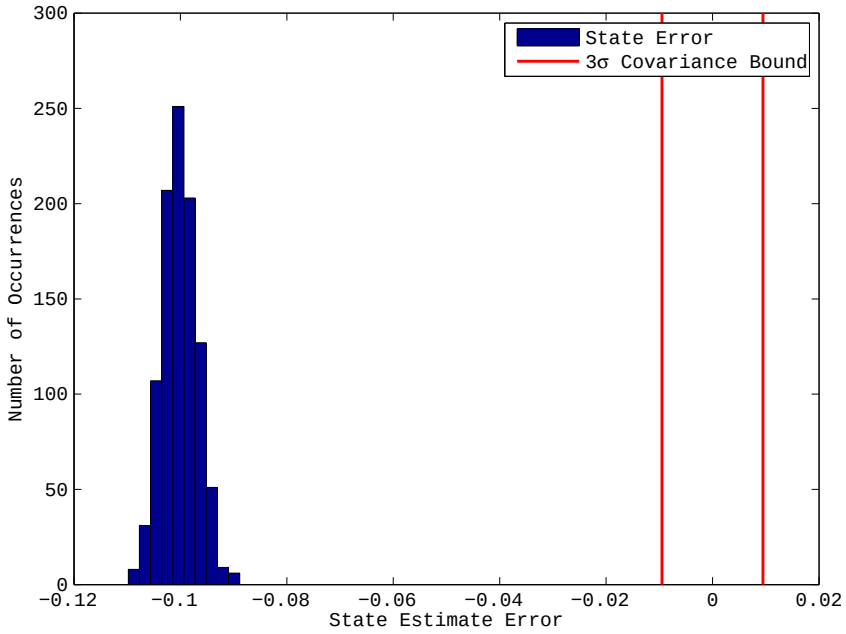


Fig. 2: Example II.1 Monte Carlo Results - TLS: No Initial Estimates

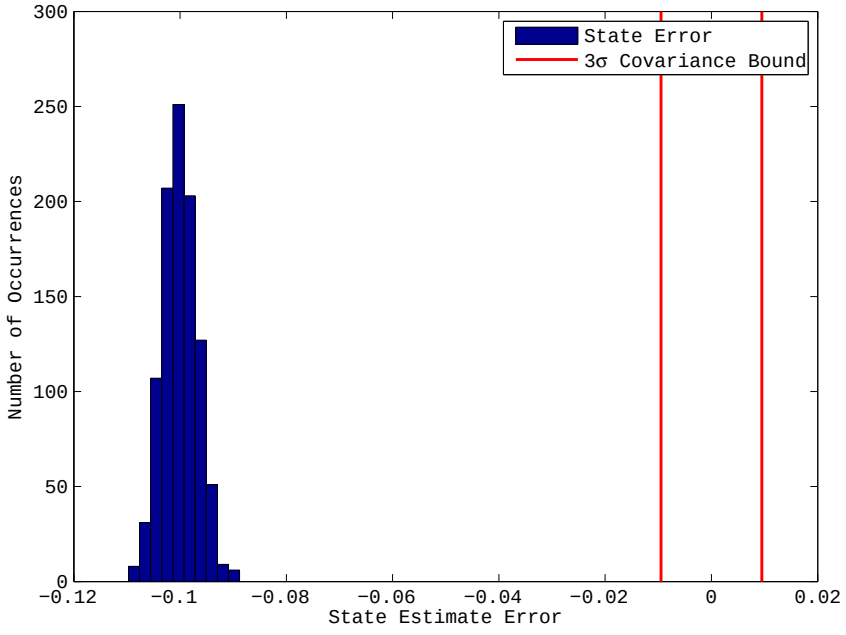


Fig. 3: Example II.1 Monte Carlo Results - TLS: State Estimates

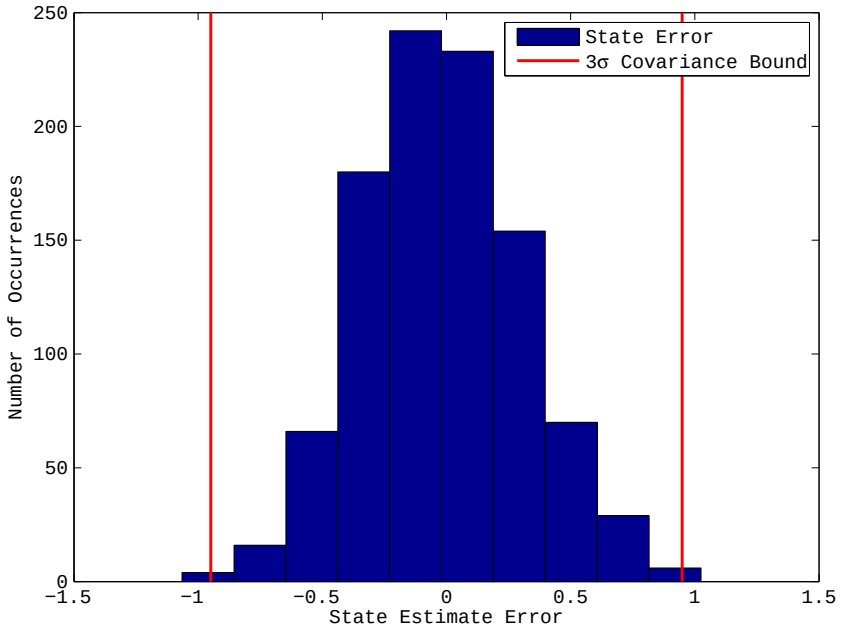


Fig. 4: Example II.1 Monte Carlo Results - CLS: No Initial Estimates

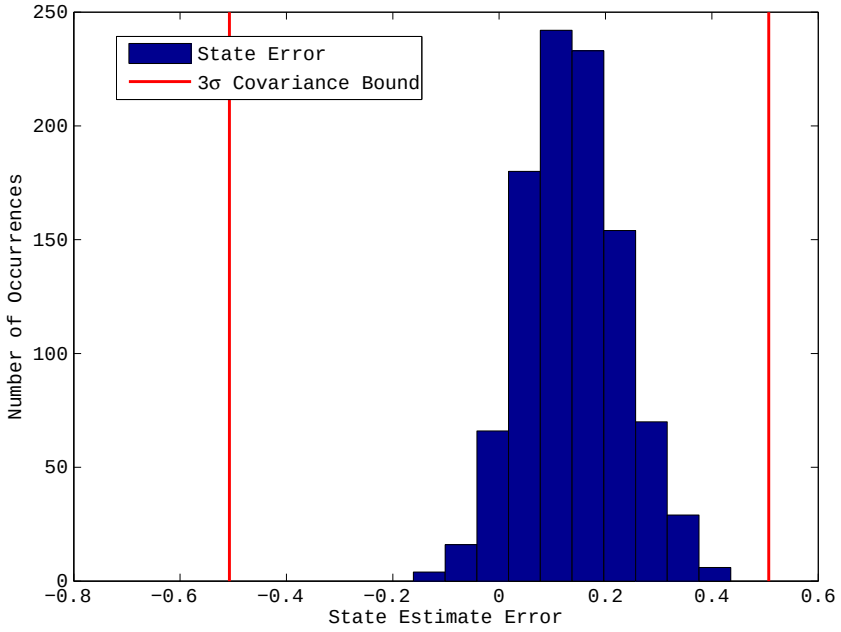


Fig. 5: Example II.1 Monte Carlo Results - CLS: State Estimates

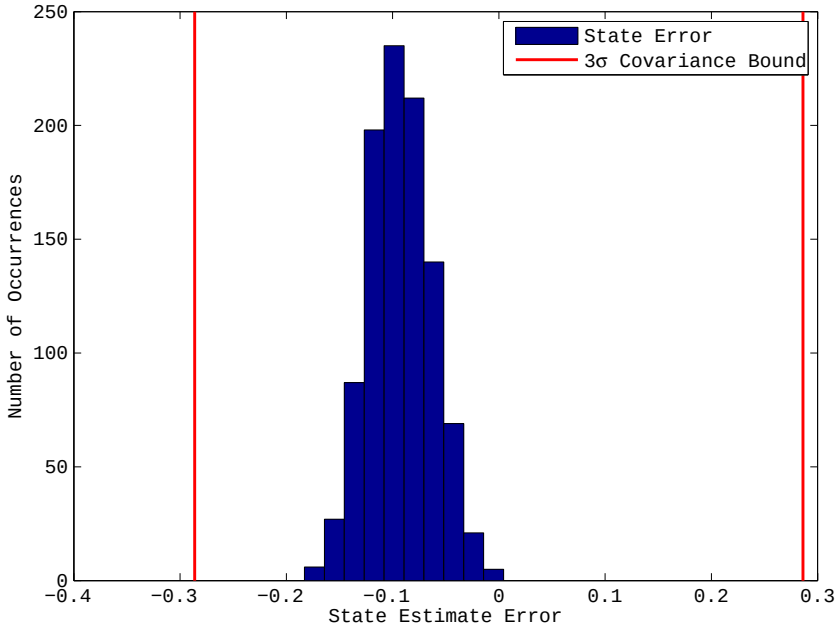


Fig. 6: Example II.1 Monte Carlo Results - CLS: Parameter Estimates

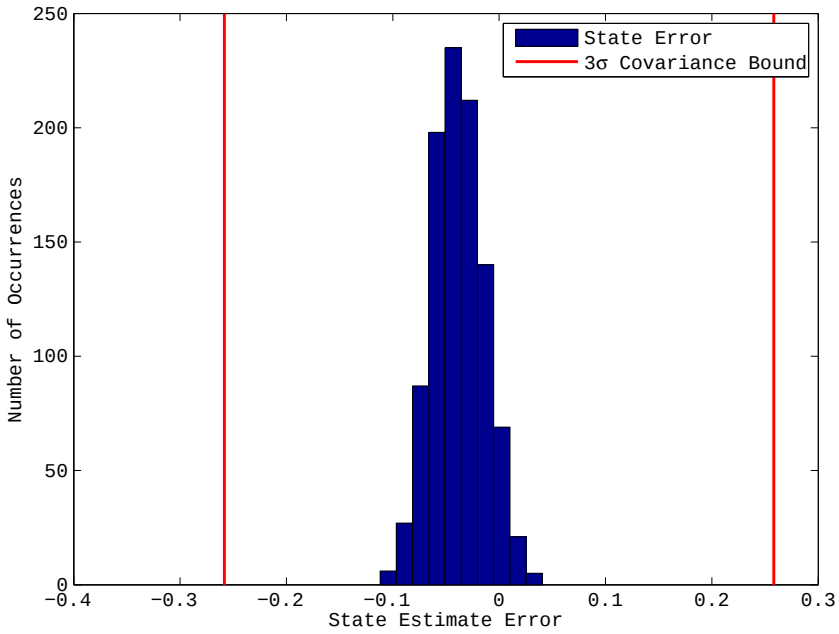


Fig. 7: Example II.1 Monte Carlo Results - TLS: State and Parameter Estimates

Since this system is Linear-Time Invariant (LTI), the State Transition Matrix (STM) is known to be

$$\Phi(t,t_0) = e^{A\Delta t} = \begin{bmatrix} \cos(\omega_n \Delta t) & \frac{1}{\omega_n}\sin(\omega_n \Delta t) \\ -\omega_n \sin(\omega_n \Delta t) & \cos(\omega_n \Delta t) \end{bmatrix} \tag{2.88}$$

where $\Delta t = t - t_0$ and

$$\boldsymbol{x}(t) = \Phi(t,t_0)\boldsymbol{x}(t_0) = \Phi(\Delta t)\boldsymbol{x}_0 \tag{2.89}$$

Because the system is dynamic, batch estimation is applied using the initial values of the states as the solve-for parameters.

Two measurements are available to monitor the states. The first is a position measurement subject to only white noise

$$\tilde{y}_1 = x_1 + v_1 \qquad v_1 \sim \mathrm{N}(0, R_1) \tag{2.90}$$

The second is a velocity measurement subject to a constant bias and white noise

$$\tilde{y}_2 = x_2 + b + v_2 \qquad v_2 \sim \mathrm{N}(0, R_2) \tag{2.91}$$

Based on these two measurements, three different scenarios are examined. The first uses only the unbiased position measurements with the Traditional Least Squares approach. The second uses both measurements, but still uses Traditional Least Squares. The third scenario uses both measurements and a Consider Least Squares framework to estimate the solution.

Given that each set of measurements is taken at equal time steps and because this is an LTI system, the measurements can be propagated by

$$\tilde{\boldsymbol{y}}_k = \begin{bmatrix} y_{1_k} \\ y_{2_k} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \Phi^k(\Delta t) \begin{bmatrix} x_{1_0} \\ x_{2_0} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} b + \begin{bmatrix} v_{1_k} \\ v_{2_k} \end{bmatrix} = H_x \Phi^k(\Delta t)\boldsymbol{x}_0 + H_p b + \boldsymbol{v}_k \tag{2.92}$$

Table 3: State and Parameter Values Used in Example II.2

Variable	x_{1_0}	x_{2_0}	b	ω_n
True	2.3	0.2	0.04	1
Estimated	3	0.0	0.08	1

Table 4: Variance Values Used in Example II.2

Variance	$P_{x_1x_{1_0}}$	$P_{x_2x_{2_0}}$	P_{pp}	R_1	R_2
Value	0.49	0.04	0.0016	0.01	0.01

The values used in this scenario are shown in Tables 3 and 4. Values not shown are set to zero. Note that the bias is within the measurement noise of the sensors.

A 1000-run Monte Carlo test was performed for all three scenarios. 100 values of each measurement were collected over a 10 second time interval for each run. *A priori* estimates were provided for both initial states as well as the bias, b. The means and standard deviations for each updated state estimate for each scenario are shown in Table 5. Comparing scenarios 1 and 2 shows that introducing the second measurement improved the resulting covariance, but introduced a bias into the estimates. Despite the measurement bias being smaller than the measurement noise, it still produced a bias in the resulting estimates. This is also shown by the graphs in Figures 8 and 9. These figures plot a histogram of the initial position estimates from each Monte Carlo run. The estimated 3σ covariance bounds have also been plotted.

The results of Scenario 3 show that using Consider Least Squares not only reduces the covariance as in Scenario 2, but the bias from the estimates is also removed. Figure 10 shows the histogram for the position estimates. Similar results were ob-

served for the initial velocity estimates, but are not as pronounced as those for the initial position and are not shown here.

Table 5: Means and Standard Deviations from a 1000-run Monte Carlo Simulation for Example II.2

		Scenario 1	**Scenario 2**	**Scenario 3**
$\boldsymbol{x_1}$	mean	-0.0002	-0.0072	0.0004
	stan. dev.	0.0131	0.0099	0.0100
$\boldsymbol{x_2}$	mean	-0.0014	-0.0027	-0.0004
	stan. dev.	0.0143	0.0102	0.0102

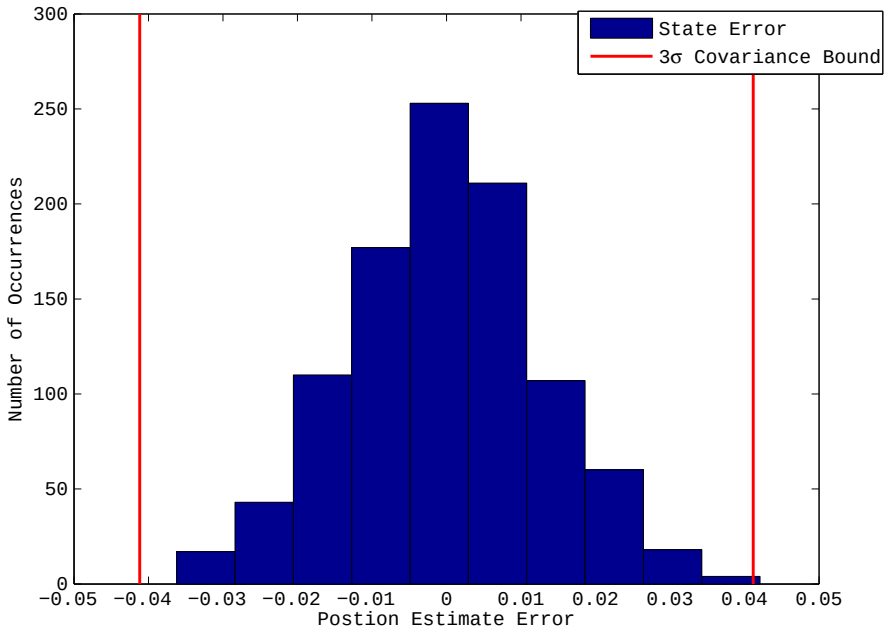


Fig. 8: Example II.2 Monte Carlo Results: Position Estimates - Scenario 1

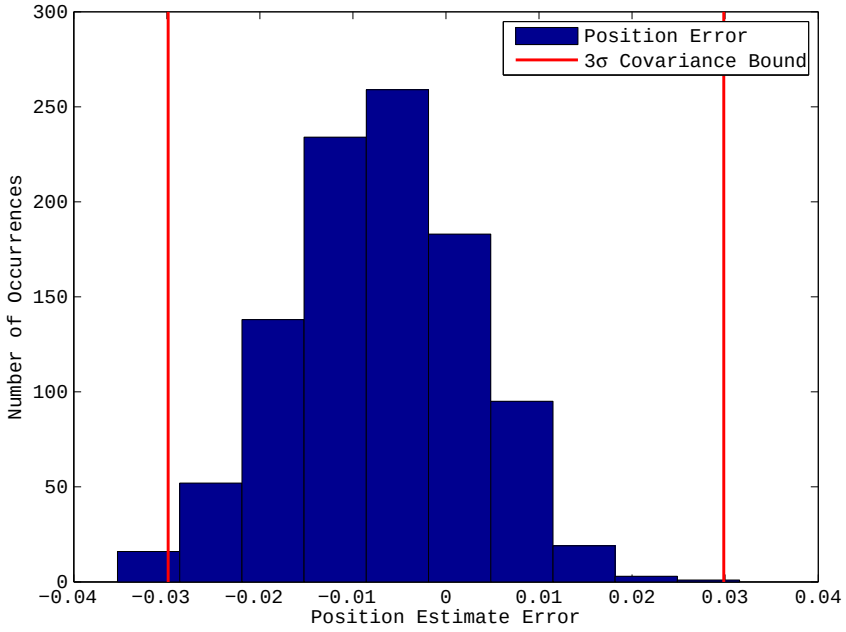


Fig. 9: Example II.2 Monte Carlo Results: Position Estimates - Scenario 2

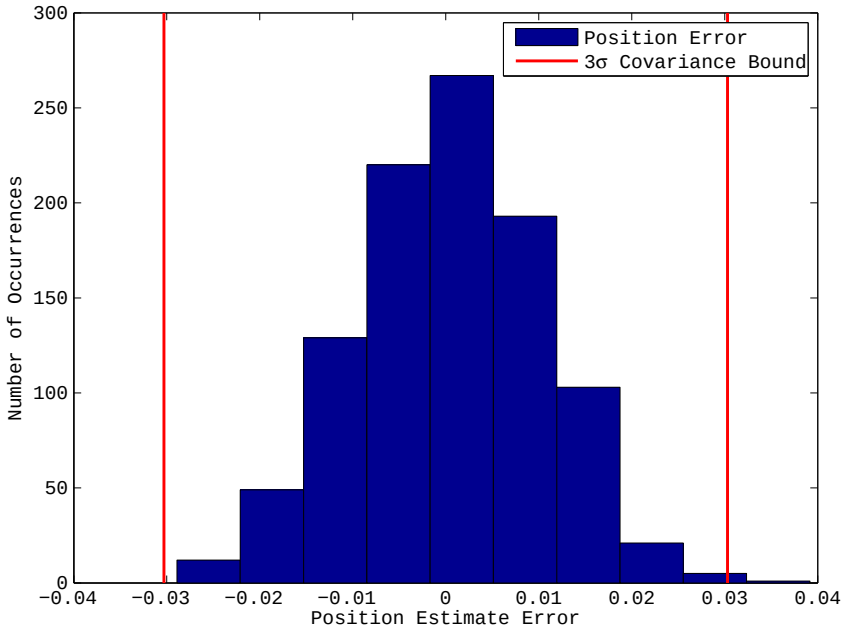


Fig. 10: Example II.2 Monte Carlo Results: Position Estimates - Scenario 3

CHAPTER III

THE CONSIDER KALMAN FILTER

In the previous chapter, methods of accounting for parameter uncertainty were explored for static systems. This chapter expands upon that knowledge by including dynamics in the system model. Only sequential estimators will be examined. This chapter will first explain the theory behind the consider Kalman filter (CKF) and its multiple derivations. A linear and nonlinear example are also explored and additional insights based on the results of the examples will be discussed.

A. Theoretical Development

This section provides the theoretical background behind the derivation and use of consider Kalman filters (CKFs). A discrete, continuous, and a continuous-discrete minimum variance consider Kalman filter (MVCKF) are developed. Even though continuous system models are no longer widely used because of digital computing, this derivation of the MVCKF is also included for the sake of completeness. Stability of both the discrete and continuous derivations are also examined. Finally, the continuous-discrete derivation is expanded to work with nonlinear systems.

1. The Discrete MVCKF

a. Measurement Model

The discrete linear measurement model is the same as that used in Chapter II, but is reproduced here for convenience

$$\tilde{\boldsymbol{y}}_k = H_{x_k}\boldsymbol{x}_k + H_{p_k}\boldsymbol{p} + \boldsymbol{v}_k, \qquad \boldsymbol{v}_k \sim \mathrm{N}(0, R_k) \tag{3.1}$$

An augmented state vector (i.e., combined state and parameter vector), $\boldsymbol{z}$, is still defined as

$$\boldsymbol{z}_k = \begin{bmatrix} \boldsymbol{x}_k \\ \boldsymbol{p} \end{bmatrix} \tag{3.2}$$

Using the augmented state vector, the measurement equation is rewritten as

$$\tilde{\boldsymbol{y}}_k = H_{z_k} \boldsymbol{z}_k + \boldsymbol{v}_k \tag{3.3}$$

where the combined measurement matrix, H_{z_k}, is still

$$H_{z_k} = \begin{bmatrix} H_{x_k} & H_{p_k} \end{bmatrix} \tag{3.4}$$

Given *a priori* estimates of both the states and parameters, it is assumed that they differ from the true values by Gaussian noise or

$$\hat{\boldsymbol{x}}^- = \boldsymbol{x} + \boldsymbol{\eta}, \qquad \boldsymbol{\eta} \sim \mathrm{N}(0, P_{xx}^-) \tag{3.5}$$

$$\hat{\boldsymbol{p}}^- = \boldsymbol{p} + \boldsymbol{\beta}, \qquad \boldsymbol{\beta} \sim \mathrm{N}(0, P_{pp}^-) \tag{3.6}$$

The cross-covariances (i.e., covariance between the states and parameters) are defined as

$$E\left\{\left(\hat{\boldsymbol{x}}^- - \boldsymbol{x}\right)\left(\hat{\boldsymbol{p}}^- - \boldsymbol{p}\right)^{\mathrm{T}}\right\} = E\left\{\boldsymbol{\eta}\boldsymbol{\beta}^{\mathrm{T}}\right\} = P_{xp}^- \tag{3.7}$$

$$E\left\{\left(\hat{\boldsymbol{p}}^- - \boldsymbol{p}\right)\left(\hat{\boldsymbol{x}}^- - \boldsymbol{x}\right)^{\mathrm{T}}\right\} = E\left\{\boldsymbol{\beta}\boldsymbol{\eta}^{\mathrm{T}}\right\} = P_{px}^- = P_{xp}^{-^{\mathrm{T}}} \tag{3.8}$$

It is also assumed that the states and parameters are uncorrelated with the measurement noise

$$E\left\{\left(\hat{\boldsymbol{x}}^- - \boldsymbol{x}\right)\boldsymbol{v}^{\mathrm{T}}\right\} = E\left\{\boldsymbol{\eta}\boldsymbol{v}^{\mathrm{T}}\right\} = 0 \tag{3.9}$$

$$E\left\{\left(\hat{\boldsymbol{p}}^- - \boldsymbol{p}\right)\boldsymbol{v}^{\mathrm{T}}\right\} = E\left\{\boldsymbol{\beta}\boldsymbol{v}^{\mathrm{T}}\right\} = 0 \tag{3.10}$$

These equations are valid assumptions since the *a priori* state and parameter estimates are not dependent on the current measurement noise values. Note that *a priori* estimates are denoted by $^-$ while *a posteriori* estimates are denoted by $^+$ as before.

Based on these definitions, we can define the augmented state covariance as

$$P_z^- = E\left\{\left(\hat{\boldsymbol{z}}^- - \boldsymbol{z}\right)\left(\hat{\boldsymbol{z}}^- - \boldsymbol{z}\right)^{\mathrm{T}}\right\} = \begin{bmatrix} P_{xx}^- & P_{xp}^- \\ P_{px}^- & P_{pp}^- \end{bmatrix} \tag{3.11}$$

b. Dynamic Model

A discrete linear dynamic system with process noise is defined as

$$\boldsymbol{x}_{k+1} = \Phi_k \boldsymbol{x}_k + \Psi_k \boldsymbol{p} + \Gamma_k \boldsymbol{w}_k, \qquad \boldsymbol{w}_k \sim \mathrm{N}(0, Q_k) \tag{3.12}$$

where $\boldsymbol{w}$ is the process noise. By once again combining the states and parameters, the augmented version of Eq. (3.12) can be written as

$$\boldsymbol{z}_{k+1} = \begin{bmatrix} \boldsymbol{x}_{k+1} \\ \boldsymbol{p} \end{bmatrix} = \begin{bmatrix} \Phi_k & \Psi_k \\ 0 & I \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_k \\ \boldsymbol{p} \end{bmatrix} + \begin{bmatrix} \Gamma_k \\ 0 \end{bmatrix} \boldsymbol{w}_k = \Theta_k \boldsymbol{z}_k + \Upsilon_k \boldsymbol{w}_k \tag{3.13}$$

where

$$\Theta_k = \begin{bmatrix} \Phi_k & \Psi_k \\ 0 & I \end{bmatrix}, \qquad \Upsilon_k = \begin{bmatrix} \Gamma_k \\ 0 \end{bmatrix} \tag{3.14}$$

When additional measurements become available, it is desirable to propagate the current state estimates, $\hat{\boldsymbol{x}}_k^+$ to the new measurement step k+1. The estimated state propagation equation is

$$\hat{\boldsymbol{z}}_{k+1}^- = \Theta_k \hat{\boldsymbol{z}}_k^+ \tag{3.15}$$

c. Discrete-Time MVCKF Derivation

Using the discrete measurement and dynamic models described above, the minimum variance consider Kalman filter is derived below. While the derivation method presented below differs from other derivations of the MVCKF, equivalent results are obtained. The stability of the filter is then analyzed theoretically.

To begin the MVCKF derivation, we seek a linear, unbiased estimator that provides minimum variance state estimates, while considering parametric errors. Thus, the updated estimate sought is a linear combination of the measurements and *a priori* estimates,

$$\begin{bmatrix} \hat{\boldsymbol{x}}^+ \\ \hat{\boldsymbol{p}}^+ \end{bmatrix} = M_1 \begin{bmatrix} \hat{\boldsymbol{x}}^- \\ \hat{\boldsymbol{p}}^- \end{bmatrix} + M_2 \tilde{\boldsymbol{y}} + M_3 \tag{3.16}$$

To make the estimator unbiased, the expected value of the update should be the true values of the states. In other words, the unbiased constraint is

$$E\left\{ \begin{bmatrix} \hat{\boldsymbol{x}}^+ \\ \hat{\boldsymbol{p}}^+ \end{bmatrix} \right\} = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{p} \end{bmatrix} = M_1 E\left\{ \begin{bmatrix} \hat{\boldsymbol{x}}^- \\ \hat{\boldsymbol{p}}^- \end{bmatrix} \right\} + M_2 E\left\{ \tilde{\boldsymbol{y}} \right\} + M_3 \tag{3.17}$$

since the expectation function is a linear operator. Using the assumption that the *a priori* state and parameter estimates differ from the truth by a zero-mean Gaussian distribution, Eqs. (3.5) and (3.6), and that the measurements are a linear combination of the true states and parameters from Eq. (3.3), the above equation becomes

$$\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{p} \end{bmatrix} = M_1 \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{p} \end{bmatrix} + M_2 \begin{bmatrix} H_x & H_p \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{p} \end{bmatrix} + M_3 \tag{3.18}$$

Thus, for this unbiased constraint condition to hold true

$$M_3 = 0, \qquad M_1 + M_2 \begin{bmatrix} H_x & H_p \end{bmatrix} = I \tag{3.19}$$

or

$$M_1 = I - M_2 \begin{bmatrix} H_x & H_p \end{bmatrix} \tag{3.20}$$

Plugging this back into the original equation results in M_2 as the only unknown:

$$\begin{bmatrix} \hat{\boldsymbol{x}}^+ \\ \hat{\boldsymbol{p}}^+ \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{x}}^- \\ \hat{\boldsymbol{p}}^- \end{bmatrix} + M_2 \left(\tilde{\boldsymbol{y}} - \begin{bmatrix} H_x & H_p \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}^- \\ \hat{\boldsymbol{p}}^- \end{bmatrix} \right) \tag{3.21}$$

In the consider methodology, parameter estimates and their associated covariance remain unchanged by implementing the constant parameter constraint (CPC). This constraint forces the *a priori* parameter estimate to be equal to its *a posteriori* estimate or

$$\hat{\boldsymbol{p}}^{+} = \hat{\boldsymbol{p}}^{-} = \hat{\boldsymbol{p}} \tag{3.22}$$

To enforce the CPC, M_2 has the structure

$$M_2 = \begin{bmatrix} K \\ \\ 0 \end{bmatrix} \tag{3.23}$$

Thus, the state update equation becomes the familiar form

$$\hat{\boldsymbol{x}}^{+} = \hat{\boldsymbol{x}}^{-} + K\left(\tilde{\boldsymbol{y}} - H_x\hat{\boldsymbol{x}}^{-} - H_p\hat{\boldsymbol{p}}\right) \tag{3.24}$$

To find the updated state covariance and optimal gain, K, the *a posteriori* state covariance is evaluated by substituting in Eqs. (3.24) and (3.1), from which it is found that

$$\begin{aligned} P_{xx}^{+} = {} & E\left\{K\left(\boldsymbol{v} - H_x\left(\hat{\boldsymbol{x}}^{-} - \boldsymbol{x}\right) - H_p\left(\hat{\boldsymbol{p}} - \boldsymbol{p}\right)\right)\left(\boldsymbol{v} - H_x\left(\hat{\boldsymbol{x}}^{-} - \boldsymbol{x}\right) - H_p\left(\hat{\boldsymbol{p}} - \boldsymbol{p}\right)\right)^{\mathrm{T}} K^{\mathrm{T}}\right\} \\ & + E\left\{\left(\hat{\boldsymbol{x}}^{-} - \boldsymbol{x}\right)\left(\boldsymbol{v} - H_x\left(\hat{\boldsymbol{x}}^{-} - \boldsymbol{x}\right) - H_p\left(\hat{\boldsymbol{p}} - \boldsymbol{p}\right)\right)^{\mathrm{T}} K^{\mathrm{T}}\right\} \\ & + E\left\{K\left(\boldsymbol{v} - H_x\left(\hat{\boldsymbol{x}}^{-} - \boldsymbol{x}\right) - H_p\left(\hat{\boldsymbol{p}} - \boldsymbol{p}\right)\right)\left(\hat{\boldsymbol{x}}^{-} - \boldsymbol{x}\right)^{\mathrm{T}}\right\} \\ & + E\left\{\left(\hat{\boldsymbol{x}}^{-} - \boldsymbol{x}\right)\left(\hat{\boldsymbol{x}}^{-} - \boldsymbol{x}\right)^{\mathrm{T}}\right\} \end{aligned} \tag{3.25}$$

where P_{xx}^{+} is defined most fundamentally as

$$P_{xx}^{+} = E\left\{\left(\hat{\boldsymbol{x}}^{+} - \boldsymbol{x}\right)\left(\hat{\boldsymbol{x}}^{+} - \boldsymbol{x}\right)^{\mathrm{T}}\right\} \tag{3.26}$$

Following some algebraic manipulation and using Eqs. (3.5)-(3.10), Eq. (3.25) is reduced to

$$P_{xx}^{+} = P_{xx}^{-} - K\left(H_x P_{xx}^{-} + H_p P_{px}^{-}\right) - \left(P_{xx}^{-} H_x^{\mathrm{T}} + P_{xp}^{-} H_p^{\mathrm{T}}\right) K^{\mathrm{T}} + K\left(H_x P_{xx}^{-} H_x^{\mathrm{T}} + H_x P_{xp}^{-} H_p^{\mathrm{T}} + H_p P_{px}^{-} H_x^{\mathrm{T}} + H_p P_{pp} H_p^{\mathrm{T}} + R\right) K^{\mathrm{T}} \tag{3.27}$$

In a similar manner it can be shown that

$$P_{xp}^{+} = E\left\{\left(\hat{\boldsymbol{x}}^{+} - \boldsymbol{x}\right)\left(\hat{\boldsymbol{p}} - \boldsymbol{p}\right)^{\mathrm{T}}\right\} = \left(I - KH_x\right) P_{xp}^{-} - KH_p P_{pp} \tag{3.28}$$

To truly call this a minimal variance solution, the optimal gain, K, must be found for the estimated states by taking the minimum of the trace of the state covariance, P_{xx}^{+}, or

$$J = \mathrm{tr}\left(P_{xx}^{+}\right) \tag{3.29}$$

Taking the partial derivative of the above cost function with respect to the gain, K, and using the trace derivative properties found in Appendix A results in

$$\frac{\partial J}{\partial K} = 0 = 2K\left(H_x P_{xx}^{-} H_x^{\mathrm{T}} + H_x P_{xp}^{-} H_p^{\mathrm{T}} + H_p P_{px}^{-} H_x^{\mathrm{T}} + H_p P_{pp} H_p^{\mathrm{T}} + R\right) - 2\left(P_{xx}^{-} H_x^{\mathrm{T}} + P_{xp}^{-} H_p^{\mathrm{T}}\right) \tag{3.30}$$

Solving for the gain produces

$$K = \left(P_{xx}^{-} H_x^{\mathrm{T}} + P_{xp}^{-} H_p^{\mathrm{T}}\right)\left(H_x P_{xx}^{-} H_x^{\mathrm{T}} + H_x P_{xp}^{-} H_p^{\mathrm{T}} + H_p P_{px}^{-} H_x^{\mathrm{T}} + H_p P_{pp} H_p^{\mathrm{T}} + R\right)^{-1} \tag{3.31}$$

Plugging this back into Eq. (3.27) yields

$$P_{xx}^{+} = \left(I - KH_x\right) P_{xx}^{-} - KH_p P_{px}^{-} \tag{3.32}$$

Note that the update Eqs. (3.24), (3.28), (3.31), and (3.32) are same as those found in the consider sequential estimator discussed in Chapter II.

d. Discrete-Time Propagation

While the states can be propagated using the estimated propagation equations, additional equations must be found to propagate the covariance and complete the filter. The propagated covariance is defined as

$$P^-_{z,k+1} = E\left\{\left(\hat{\boldsymbol{z}}^-_{k+1} - \boldsymbol{z}_{k+1}\right)\left(\hat{\boldsymbol{z}}^-_{k+1} - \boldsymbol{z}_{k+1}\right)^{\mathrm{T}}\right\} \tag{3.33}$$

By plugging in Eqs. (3.13) and (3.15)

$$P^-_{z,k+1} = E\left\{\left[\Theta_k\left(\hat{\boldsymbol{z}}^+_k - \boldsymbol{z}_k\right) - \Upsilon_k \boldsymbol{w}_k\right]\left[\Theta_k\left(\hat{\boldsymbol{z}}^+_k - \boldsymbol{z}_k\right) - \Upsilon_k \boldsymbol{w}_k\right]^{\mathrm{T}}\right\} \tag{3.34}$$

Using the assumption that the states and parameters are uncorrelated with the process noise, expanding the equation above yields the propagated covariance

$$P^-_{z,k+1} = \Theta_k P^+_{z,k} \Theta^{\mathrm{T}}_k + \Upsilon_k Q_k \Upsilon^{\mathrm{T}}_k \tag{3.35}$$

where $P^+_{z,k}$ is the updated or *a posteriori* covariance at step k.

Furthermore, since

$$P^-_{z,k+1} = \begin{bmatrix} P^-_{xx,k+1} & P^-_{xp,k+1} \\ P^-_{px,k+1} & P^-_{pp,k+1} \end{bmatrix} \tag{3.36}$$

the component covariance matrices can be defined as

$$P^-_{xx,k+1} = \Phi_k P^+_{xx,k} \Phi^{\mathrm{T}}_k + \Phi_k P^+_{xp,k} \Psi^{\mathrm{T}}_k + \Psi_k P^+_{px,k} \Phi^{\mathrm{T}}_k + \Psi_k P^+_{pp,k} \Psi^{\mathrm{T}}_k + \Gamma_k Q_k \Gamma^{\mathrm{T}}_k \tag{3.37}$$

$$P^-_{xp,k+1} = \Phi_k P^+_{xp,k} + \Psi_k P^+_{pp,k} \tag{3.38}$$

$$P^-_{px,k+1} = P^+_{px,k} \Phi^{\mathrm{T}}_k + P^+_{pp,k} \Psi^{\mathrm{T}}_k = P^{-^{\mathrm{T}}}_{xp,k+1} \tag{3.39}$$

$$P^-_{pp,k+1} = P^+_{pp,k} = P_{pp} \tag{3.40}$$

Note that Eq. (3.40) is consistent with the initial assumption of invariant parameter covariance as specified by the CPC. The fully discrete MVCKF is summarized in

Table 6. Although a different derivation process was used, the resulting filter is equivalent to that found by Jazwinski [19].

Note that the same parameter vector is used in both the dynamic and measurement models. As a result, it is assumed that the parameter vector consists of both dynamic parameters and measurement parameters where their influence is controlled by their associated sensitivity matrices, Ψ_k and H_p, respectively. A separate derivation can be performed using different dynamic and measurement parameter vectors and is summarized by Schmidt [18]. Although only minor theoretical improvements are made by separating the dynamic and measurement model parameters, computational loads can be further reduced. The vast majority of dynamic parameters have no correlation to the measurement noise and the measurement parameters allowing the update of the dynamic parameters' associated cross-covariance matrix to be neglected. Similarly, the cross-covariance matrix for the measurement parameters will only need to be propagated in very rare cases.

e. Stability of the Discrete MVCKF

The stability of the Discrete MVCKF can be verified using Lyapunov's direct method as described by Franklin *et al.* [27]. The analysis begins using a proposed Lyapunov equation of

$$V\left(\tilde{\boldsymbol{z}}\right) = \tilde{\boldsymbol{z}}_k^{\mathrm{T}} P_{z,k}^{-1} \tilde{\boldsymbol{z}}_k \tag{3.41}$$

where $\tilde{\boldsymbol{z}}_k = \hat{\boldsymbol{z}}_k - \boldsymbol{z}_k$. To verify the system is stable, the change in the Lyapunov function, ΔV, must be at least negative semi-definite. The change in the proposed Lyapunov function is defined as

$$\Delta V\left(\tilde{\boldsymbol{z}}\right) = \tilde{\boldsymbol{z}}_{k+1}^{\mathrm{T}} P_{z,k+1}^{-1} \tilde{\boldsymbol{z}}_{k+1} - \tilde{\boldsymbol{z}}_k^{\mathrm{T}} P_{z,k}^{-1} \tilde{\boldsymbol{z}}_k \tag{3.42}$$

Table 6: Discrete Minimum Variance Consider Kalman Filter

Model	$\boldsymbol{x}_{k+1} = \Phi_k \boldsymbol{x}_k + \Psi_k \boldsymbol{p} + \Gamma_k \boldsymbol{w}_k, \qquad \boldsymbol{w}_k \sim \mathrm{N}(0, Q_k)$ $\tilde{\boldsymbol{y}}_k = H_{x_k} \boldsymbol{x}_k + H_{p_k} \boldsymbol{p} + \boldsymbol{v}_k, \qquad \boldsymbol{v}_k \sim \mathrm{N}(0, R_k)$
Gain	$K = \left(P_{xx}^- H_x^{\mathrm{T}} + P_{xp}^- H_p^{\mathrm{T}}\right)$ $\times \left(H_x P_{xx}^- H_x^{\mathrm{T}} + H_x P_{xp}^- H_p^{\mathrm{T}} + H_p P_{px}^- H_x^{\mathrm{T}} + H_p P_{pp} H_p^{\mathrm{T}} + R\right)^{-1}$
Update	$\hat{\boldsymbol{x}}^+ = \hat{\boldsymbol{x}}^- + K\left(\tilde{\boldsymbol{y}} - H_x \hat{\boldsymbol{x}}^- - H_p \hat{\boldsymbol{p}}\right)$ $P_{xx}^+ = \left(I - K H_x\right) P_{xx}^- - K H_p P_{px}^-$ $P_{xp}^+ = \left(I - K H_x\right) P_{xp}^- - K H_p P_{pp}$
Propagation	$\hat{\boldsymbol{x}}_{k+1}^- = \Phi_k \hat{\boldsymbol{x}}_k^+ + \Psi_k \hat{\boldsymbol{p}}$ $P_{xx,k+1}^- = \Phi_k P_{xx,k}^+ \Phi_k^{\mathrm{T}} + \Phi_k P_{xp,k}^+ \Psi_k^{\mathrm{T}} + \Psi_k P_{px,k}^+ \Phi_k^{\mathrm{T}}$ $+ \Psi_k P_{pp} \Psi_k^{\mathrm{T}} + \Gamma_k Q_k \Gamma_k^{\mathrm{T}}$ $P_{xp,k+1}^- = \Phi_k P_{xp,k}^+ + \Psi_k P_{pp}$

The first step in the stability analysis is to find a definition for $\tilde{\boldsymbol{z}}_{k+1}$. Using Eqs. (3.15), (3.13), (2.64), and (3.1) it is found to be

$$\tilde{\boldsymbol{z}}_{k+1} = \hat{\boldsymbol{z}}_{k+1} - \boldsymbol{z}_{k+1} = \Theta_k \left(I - K_z H_z\right) \tilde{\boldsymbol{z}}_k + \Theta_k K_z \boldsymbol{v}_k - \Upsilon_k \boldsymbol{w}_k \tag{3.43}$$

Stability is only affected by the coefficients acting on the states, thus the last two terms can be neglected. Plugging this result back into Eq. (3.42) and consolidating terms yields

$$\Delta V\left(\tilde{\boldsymbol{z}}\right) = \tilde{\boldsymbol{z}}_k^{\mathrm{T}} \left[\left(I - K_z H_z\right)^{\mathrm{T}} \Theta_k^{\mathrm{T}} P_{z,k+1}^{-1} \Theta_k \left(I - K_z H_z\right) - P_{z,k}^{-1}\right] \tilde{\boldsymbol{z}}_k \tag{3.44}$$

Since the definiteness of the matrix is being evaluated, only the values contained in the brackets need to be analyzed. Thus, for the discrete MVCKF to be stable

$$\left(I - K_z H_z\right)^{\mathrm{T}} \Theta_k^{\mathrm{T}} P_{z,k+1}^{-1} \Theta_k \left(I - K_z H_z\right) - P_{z,k}^{-1} \leq 0 \tag{3.45}$$

where the notation ≤ 0 implies the matrix is negative semi-definite. Rearranging the above equation assuming that the appropriate inverses exist gives

$$I - P_{z,k+1} \Theta_k^{-\mathrm{T}} \left(I - K_z H_z\right)^{-\mathrm{T}} P_{z,k}^{-1} \left(I - K_z H_z\right)^{-1} \Theta_k^{-1} \leq 0 \tag{3.46}$$

The quadratic or *Joseph's form* of the augmented state update covariance is given by [26]

$$P_z^+ = \left(I - K_z H_z\right) P_z^- \left(I - K_z H_z\right)^{\mathrm{T}} + K_z R K_z^{\mathrm{T}} \tag{3.47}$$

Using this result in Eq. (3.35) yields

$$P_{z,k+1} = \Theta_k \left(I - K_z H_z\right) P_z^- \left(I - K_z H_z\right)^{\mathrm{T}} \Theta_k^{\mathrm{T}} + \Theta_k K_z R K_z^{\mathrm{T}} \Theta_k^{\mathrm{T}} + \Upsilon_k Q_k \Upsilon_k^{\mathrm{T}} \tag{3.48}$$

Plugging this back into Eq. (3.46) and rearranging terms results in

$$-\left[\Theta_k K_z R K_z^{\mathrm{T}} \Theta_k^{\mathrm{T}} + \Upsilon_k Q_k \Upsilon_k^{\mathrm{T}}\right] \left[\Theta_k^{-\mathrm{T}} \left(I - K_z H_z\right)^{-\mathrm{T}} P_{z,k}^{-1} \left(I - K_z H_z\right)^{-1} \Theta_k^{-1}\right] \leq 0 \tag{3.49}$$

Since $P_{z,k}$ is positive definite, its inverse is also positive definite. Furthermore, if a matrix A is positive definite any quadratic form of A is also positive definite [13]. Thus, the second set of brackets in Eq. (3.49) is positive definite and the equation reduces to

$$-\left[\Theta_k K_z R K_z^{\mathrm{T}} \Theta_k^{\mathrm{T}} + \Upsilon_k Q_k \Upsilon_k^{\mathrm{T}}\right] \leq 0 \tag{3.50}$$

For an augmented filter the stability analysis is complete, but additional substitutions must be made for stability analysis of the MVCKF. Plugging in the definitions of Θ_k, Υ_k, and $K_z = \begin{bmatrix} K^{\mathrm{T}} & 0 \end{bmatrix}^{\mathrm{T}}$ gives

$$-\begin{bmatrix} \Phi_k K R K^{\mathrm{T}} \Phi_k^{\mathrm{T}} + \Gamma_k Q_k \Gamma_k^{\mathrm{T}} & 0 \\ 0 & 0 \end{bmatrix} \leq 0 \tag{3.51}$$

Provided that the measurement covariance, R, is positive definite and the process noise covariance, Q, is at least positive semi-definite, the discrete MVCKF will be stable, but not asymptotically stable.

2. The Continuous MVCKF

This section derives the minimum variance consider Kalman filter using fully continuous dynamic and measurement models. It is initially assumed that the parameters, $\boldsymbol{p}$, do not vary with time in accordance with the CPC. Leondes' compilation contains an alternate derivation than the one provided below [29].

Given a completely continuous linear dynamic system of equations

$$\begin{bmatrix} \dot{\boldsymbol{x}}(t) \\ \dot{\boldsymbol{p}} \end{bmatrix} = \begin{bmatrix} F(t) & B(t) \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(t) \\ \boldsymbol{p} \end{bmatrix} + \begin{bmatrix} G(t) \\ 0 \end{bmatrix} \boldsymbol{w}(t), \qquad \boldsymbol{w}(t) \sim \mathrm{N}(0, Q(t)) \tag{3.52}$$

$$\tilde{\boldsymbol{y}}(t) = H_x(t)\boldsymbol{x}(t) + H_p(t)\boldsymbol{p} + \boldsymbol{v}(t), \qquad \boldsymbol{v}(t) \sim \mathrm{N}(0, R(t)) \tag{3.53}$$

the estimated state equations can be written as

$$\begin{bmatrix} \dot{\hat{\boldsymbol{x}}}(t) \\ \dot{\hat{\boldsymbol{p}}} \end{bmatrix} = \begin{bmatrix} F(t) & B(t) \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}(t) \\ \hat{\boldsymbol{p}} \end{bmatrix} + \begin{bmatrix} K(t) \\ 0 \end{bmatrix} \left(\tilde{\boldsymbol{y}}(t) - H_x(t)\hat{\boldsymbol{x}}(t) - H_p(t)\hat{\boldsymbol{p}}\right) \tag{3.54}$$

The augmented error variable is defined as

$$\tilde{\boldsymbol{z}}(t) = \hat{\boldsymbol{z}}(t) - \boldsymbol{z}(t) = \begin{bmatrix} \hat{\boldsymbol{x}}(t) \\ \hat{\boldsymbol{p}} \end{bmatrix} - \begin{bmatrix} \boldsymbol{x}(t) \\ \boldsymbol{p} \end{bmatrix} \tag{3.55}$$

Differentiating Eq.(3.55) and substituting in Eqs. (3.52) and (3.54) gives a new set of estimation error dynamic equations

$$\dot{\tilde{\boldsymbol{z}}}(t) = \left(F_z(t) - K_z(t)H_z(t)\right)\tilde{\boldsymbol{z}}(t) - G_z(t)\boldsymbol{w}(t) + K_z(t)\boldsymbol{v}(t) \tag{3.56}$$

where

$$F_z(t) = \begin{bmatrix} F(t) & B(t) \\ 0 & 0 \end{bmatrix}, \quad K_z(t) = \begin{bmatrix} K(t) \\ 0 \end{bmatrix} \tag{3.57}$$

$$H_z(t) = \begin{bmatrix} H_x(t) & H_p(t) \end{bmatrix}, \quad G_z(t) = \begin{bmatrix} G(t) \\ 0 \end{bmatrix} \tag{3.58}$$

This new set of equations allows the use of the original continuous Kalman filter derivation as described by Crassidis and Junkins [13]. Thus, the continuous covariance propagation equation obeys the differential equation

$$\begin{aligned} \dot{P}_z(t) = {} & \left(F_z(t) - K_z(t)H_z(t)\right)P_z(t) + P_z(t)\left(F_z(t) - K_z(t)H_z(t)\right)^{\mathrm{T}} \\ & + K_z(t)R(t)K_z(t)^{\mathrm{T}} + G_z(t)Q(t)G_z(t)^{\mathrm{T}} \end{aligned} \tag{3.59}$$

By expanding out Eq. (3.59), the state covariance rate can then be brought to the form

$$\begin{aligned} \dot{P}_{xx}(t) = {} & \left(F(t) - K(t)H_x(t)\right)P_{xx}(t) + P_{xx}(t)\left(F^{\mathrm{T}}(t) - H_x^{\mathrm{T}}(t)K^{\mathrm{T}}(t)\right) \\ & + \left(B(t) - K(t)H_p(t)\right)P_{px}(t) + P_{xp}(t)\left(B^{\mathrm{T}}(t) - H_p^{\mathrm{T}}(t)K^{\mathrm{T}}(t)\right) \\ & + K(t)R(t)K^{\mathrm{T}}(t) + G(t)Q(t)G^{\mathrm{T}}(t) \end{aligned} \tag{3.60}$$

By minimizing the trace of Eq. (3.60) as in Section 3, the gain, $K(t)$, is found to be

$$K(t) = \left(P_{xx}(t)H_x^{\mathrm{T}}(t) + P_{xp}(t)H_p^{\mathrm{T}}(t)\right) R^{-1}(t) \tag{3.61}$$

Plugging this back into Eq. (3.60) and rearranging results in the modified Riccati equations

$$\begin{aligned}\dot{P}_{xx}(t) = {} & F(t)P_{xx}(t) + P_{xx}(t)F^{\mathrm{T}}(t) + B(t)P_{px}(t) + P_{xp}(t)B^{\mathrm{T}}(t) + G(t)Q(t)G^{\mathrm{T}}(t) \\ & - \left(P_{xx}(t)H_x^{\mathrm{T}}(t) + P_{xp}(t)H_p^{\mathrm{T}}(t)\right) R^{-1}(t) \left(H_x(t)P_{xx}(t) + H_p(t)P_{px}(t)\right)\end{aligned} \tag{3.62}$$

and similarly,

$$\begin{aligned}\dot{P}_{xp}(t) = {} & F(t)P_{xp}(t) + B(t)P_{pp} \\ & - \left(P_{px}(t)H_x^{\mathrm{T}}(t) + P_{pp}H_p^{\mathrm{T}}(t)\right) R^{-1}(t) \left(H_x(t)P_{xp}(t) + H_p(t)P_{pp}\right)\end{aligned} \tag{3.63}$$

The continuous MVCKF is summarized in Table 7.

a. Stability of the Continuous MVCKF

Using similar methods as those used to verify stability for the discrete MVCKF, the stability of the continuous MVCKF can also be proved. Using the candidate Lyapunov function

$$V\left(\tilde{\boldsymbol{z}}(t)\right) = \tilde{\boldsymbol{z}}^{\mathrm{T}}(t)P_z^{-1}(t)\tilde{\boldsymbol{z}}(t) \tag{3.64}$$

According to Lyapunov's direct method, the above function is stable if its derivative is negative semi-definite or [30]

$$\dot{V}\left(\tilde{\boldsymbol{z}}(t)\right) = \dot{\tilde{\boldsymbol{z}}}^{\mathrm{T}}(t)P_z^{-1}(t)\tilde{\boldsymbol{z}}(t) + \tilde{\boldsymbol{z}}^{\mathrm{T}}(t)\dot{P}_z^{-1}(t)\tilde{\boldsymbol{z}}(t) + \tilde{\boldsymbol{z}}^{\mathrm{T}}(t)P_z^{-1}(t)\dot{\tilde{\boldsymbol{z}}}(t) \tag{3.65}$$

Eq. (3.56) provides a value for $\dot{\tilde{\boldsymbol{z}}}^{\mathrm{T}}(t)$, but once again only the coefficients acting on $\tilde{\boldsymbol{z}}(t)$ are important and its derivative is simplified to

$$\dot{\tilde{\boldsymbol{z}}}(t) = \left(F_z(t) - K_z(t)H_z(t)\right)\tilde{\boldsymbol{z}}(t) \tag{3.66}$$

Table 7: Continuous Minimum Variance Consider Kalman Filter

Model	$\dot{\boldsymbol{x}}(t) = F(t)\boldsymbol{x}(t) + B(t)\boldsymbol{p} + G(t)\boldsymbol{w}(t), \quad \boldsymbol{w}(t) \sim \mathrm{N}(0, Q(t))$ $\tilde{\boldsymbol{y}}(t) = H_x(t)\boldsymbol{x}(t) + H_p(t)\boldsymbol{p} + \boldsymbol{v}(t), \quad \boldsymbol{v}(t) \sim \mathrm{N}(0, R(t))$
Gain	$K(t) = \left(P_{xx}(t)H_x^{\mathrm{T}}(t) + P_{xp}(t)H_p^{\mathrm{T}}(t)\right) R^{-1}(t)$
State	$\dot{\hat{\boldsymbol{x}}}(t) = F(t)\hat{\boldsymbol{x}}(t) + B(t)\hat{\boldsymbol{p}} + K(t)\left(\tilde{\boldsymbol{y}}(t) - H_x(t)\hat{\boldsymbol{x}}(t) - H_p(t)\hat{\boldsymbol{p}}\right)$
Covariance	$\dot{P}_{xx}(t) = F(t)P_{xx}(t) + P_{xx}(t)F^{\mathrm{T}}(t) + P_{xp}(t)B^{\mathrm{T}}(t) + B(t)P_{px}(t)$ $+G(t)Q(t)G^{\mathrm{T}}(t) - \left(P_{xx}(t)H_x^{\mathrm{T}}(t) + P_{xp}(t)H_p^{\mathrm{T}}(t)\right)$ $\times R^{-1}(t)\left(H_x(t)P_{xx}(t) + H_p(t)P_{px}(t)\right)$ $\dot{P}_{xp}(t) = F(t)P_{xp}(t) + B(t)P_{pp}$ $-\left(P_{px}(t)H_x^{\mathrm{T}}(t) + P_{pp}H_p^{\mathrm{T}}(t)\right) R^{-1}(t)\left(H_x(t)P_{xp}(t) + H_p(t)P_{pp}\right)$

Therefore, only an equation for $\dot{P}_z^{-1}(t)$ is required. This can be found by taking the derivative of the definition of an inverse or

$$\frac{d}{dt}\left(P_z(t)P_z^{-1}(t)\right) = \dot{P}_z(t)P_z^{-1}(t) + P_z(t)\dot{P}_z^{-1}(t) \tag{3.67}$$

After rearranging terms

$$\dot{P}_z^{-1}(t) = -P_z^{-1}(t)\dot{P}_z(t)P_z^{-1}(t) \tag{3.68}$$

The augmented covariance propagation equation is known to be the Riccati equation [13]

$$\dot{P}_z(t) = F_z(t)P_z(t) + P_z(t)F_z^{\mathrm{T}}(t) - P_z(t)H_z^{\mathrm{T}}(t)R^{-1}(t)H_z(t)P_z(t) + G_z(t)Q(t)G_z^{\mathrm{T}}(t) \tag{3.69}$$

with a gain, $K_z(t)$, defined as

$$K_z(t) = P_z(t)H_z^{\mathrm{T}}(t)R^{-1}(t) \tag{3.70}$$

Substituting Eqs. (3.66) and (3.68)-(3.70) into Eq. (3.65) and simplifying gives

$$-\tilde{\boldsymbol{z}}^{\mathrm{T}}(t)P_z^{-1}(t)\left[K_z(t)R(t)K_z^{\mathrm{T}}(t) + G_z(t)Q(t)G_z^{\mathrm{T}}(t)\right]P_z^{-1}(t)\tilde{\boldsymbol{z}}(t) \leq 0 \tag{3.71}$$

Since only the definiteness of the above equation is of concern, the terms outside of the brackets can be neglected. Plugging in the definitions for $K_z(t)$ and $G_z(t)$ yields

$$-\begin{bmatrix} K(t)R(t)K^{\mathrm{T}}(t) + G_z(t)Q(t)G_z^{\mathrm{T}}(t) & 0 \\ 0 & 0 \end{bmatrix} \leq 0 \tag{3.72}$$

Thus, if $R(t)$ is positive definite and $Q(t)$ is at least positive semi-definite then the continuous form of the MVCKF will always be stable, but not asymptotically stable just as for the discrete form.

3. Continuous-discrete MVCKF Derivation

In a majority of applications the dynamics of a system are represented by a continuous time model while the measurements are sampled discretely via a digital computer. As a result, a hybrid filter must be developed to account for this situation. Fortunately, the discrete update equations can still be used for the hybrid filter's update step, meaning that only a new covariance propagation equation needs to be developed.

The true continuous linear dynamical model with process noise is still defined as it was in Eq. (3.52)

$$\dot{\boldsymbol{x}}(t) = F(t)\boldsymbol{x}(t) + B(t)\boldsymbol{p} + G(t)\boldsymbol{w}(t), \qquad \boldsymbol{w} \sim N(0, Q(t)) \tag{3.73}$$

The solution to this dynamic model can be written as [8, 13]

$$\boldsymbol{x}(t) = \Phi(t, t_0)\boldsymbol{x}(t_0) + \Psi(t, t_0)\boldsymbol{p} + \int_{t_0}^{t} \Phi(t, \tau)G(\tau)\boldsymbol{w}(\tau)d\tau \tag{3.74}$$

where

$$\Phi(t, t_j) = \frac{\partial \boldsymbol{x}(t)}{\partial \boldsymbol{x}(t_j)} \tag{3.75}$$

$$\Psi(t, t_j) = \frac{\partial \boldsymbol{x}(t)}{\partial \boldsymbol{p}} \tag{3.76}$$

with the following conditions on their derivatives and initial conditions

$$\dot{\Phi}(t, t_j) = F(t)\Phi(t, t_j), \qquad \Phi(t_j, t_j) = I \tag{3.77}$$

and

$$\dot{\Psi}(t, t_j) = F(t)\Psi(t, t_j) + B(t), \qquad \Psi(t_j, t_j) = 0 \tag{3.78}$$

The estimated state propagation equation is defined as

$$\dot{\hat{\boldsymbol{x}}}(t) = F(t)\hat{\boldsymbol{x}}(t) + B(t)\hat{\boldsymbol{p}} \tag{3.79}$$

with solution

$$\hat{\boldsymbol{x}}(t) = \Phi(t, t_0)\hat{\boldsymbol{x}}(t_0) + \Psi(t, t_0)\hat{\boldsymbol{p}} \tag{3.80}$$

which is slightly different than the one defined by the completely continuous derivation.

Using the solution of $\dot{\hat{\boldsymbol{x}}}(t)$ in Eq. (3.80), the state covariance is

$$\begin{aligned} P_{xx}(t) &= E\left\{(\hat{\boldsymbol{x}}(t) - \boldsymbol{x}(t))\,(\hat{\boldsymbol{x}}(t) - \boldsymbol{x}(t))^{\mathrm{T}}\right\} \\ &= \Phi(t, t_0)P_{xx}(t_0)\Phi^{\mathrm{T}}(t, t_0) + \Phi(t, t_0)P_{xp}(t_0)\Psi^{\mathrm{T}}(t, t_0) + \Psi(t, t_0)P_{px}(t_0)\Phi^{\mathrm{T}}(t, t_0) \\ &\quad + \Psi(t, t_0)P_{pp}\Psi^{\mathrm{T}}(t, t_0) + \int_{t_0}^{t} \Phi(t, \tau)G(\tau)Q(\tau)G^{\mathrm{T}}(\tau)\Phi^{\mathrm{T}}(t, \tau)d\tau \end{aligned} \tag{3.81}$$

Taking its derivative yields

$$\begin{aligned} \dot{P}_{xx}(t) &= \dot{\Phi}(t, t_0)P_{xx}(t_0)\Phi^{\mathrm{T}}(t, t_0) + \Phi(t, t_0)P_{xx}(t_0)\dot{\Phi}^{\mathrm{T}}(t, t_0) + \dot{\Phi}(t, t_0)P_{xp}(t_0)\Psi^{\mathrm{T}}(t, t_0) \\ &\quad + \Phi(t, t_0)P_{xp}(t_0)\dot{\Psi}^{\mathrm{T}}(t, t_0) + \dot{\Psi}(t, t_0)P_{px}(t_0)\Phi^{\mathrm{T}}(t, t_0) \\ &\quad + \Psi(t, t_0)P_{px}(t_0)\dot{\Phi}^{\mathrm{T}}(t, t_0) + \dot{\Psi}(t, t_0)P_{pp}\Psi^{\mathrm{T}}(t, t_0) + \Psi(t, t_0)P_{pp}\dot{\Psi}^{\mathrm{T}}(t, t_0) \\ &\quad + \int_{t_0}^{t} \dot{\Phi}(t, \tau)G(\tau)Q(\tau)G^{\mathrm{T}}(\tau)\Phi^{\mathrm{T}}(t, \tau)d\tau \\ &\quad + \int_{t_0}^{t} \Phi(t, \tau)G(\tau)Q(\tau)G^{\mathrm{T}}(\tau)\dot{\Phi}^{\mathrm{T}}(t, \tau)d\tau + \Phi(t, t)G(t)Q(t)G^{\mathrm{T}}(t)\Phi^{\mathrm{T}}(t, t) \end{aligned} \tag{3.82}$$

Collecting terms after using the definitions of $\dot{\Phi}(t,t_0)$, $\dot{\Psi}(t,t_0)$, and $\Phi(t,t)$ produces

$$\begin{aligned}\dot{P}_{xx}(t) = F(t)\Bigg(&\Phi(t,t_0)P_{xx}(t_0)\Phi^{\mathrm{T}}(t,t_0) + \Phi(t,t_0)P_{xp}(t_0)\Psi^{\mathrm{T}}(t,t_0)\\ &+ \Psi(t,t_0)P_{px}(t_0)\Phi^{\mathrm{T}}(t,t_0) + \Psi(t,t_0)P_{pp}\Psi^{\mathrm{T}}(t,t_0)\\ &+ \int_{t_0}^{t}\Phi(t,\tau)G(\tau)Q(\tau)G^{\mathrm{T}}(\tau)\Phi^{\mathrm{T}}(t,\tau)d\tau\Bigg)\\ &+ \Bigg(\Phi(t,t_0)P_{xx}(t_0)\Phi^{\mathrm{T}}(t,t_0) + \Phi(t,t_0)P_{xp}(t_0)\Psi^{\mathrm{T}}(t,t_0)\\ &+ \Psi(t,t_0)P_{px}(t_0)\Phi^{\mathrm{T}}(t,t_0) + \Psi(t,t_0)P_{pp}\Psi^{\mathrm{T}}(t,t_0)\\ &+ \int_{t_0}^{t}\Phi(t,\tau)G(\tau)Q(\tau)G^{\mathrm{T}}(\tau)\Phi^{\mathrm{T}}(t,\tau)d\tau\Bigg)F^{\mathrm{T}}(t)\\ &+ B(t)P_{px}(t_0)\Phi^{\mathrm{T}}(t,t_0) + \Phi(t,t_0)P_{xp}(t_0)B^{\mathrm{T}}(t)\\ &+ B(t)P_{pp}\Psi^{\mathrm{T}}(t,t_0) + \Psi(t,t_0)P_{pp}B^{\mathrm{T}}(t) + G(t)Q(t)G^{\mathrm{T}}(t)\end{aligned} \tag{3.83}$$

Plugging in the definition of $P_{xx}(t)$ results in further simplification

$$\begin{aligned}\dot{P}_{xx}(t) = F(t)P_{xx}(t) + P_{xx}(t)F^{\mathrm{T}}(t) + B(t)P_{px}(t_0)\Phi^{\mathrm{T}}(t,t_0)\\ + \Phi(t,t_0)P_{xp}(t_0)B^{\mathrm{T}}(t) + B(t)P_{pp}\Psi^{\mathrm{T}}(t,t_0)\\ + \Psi(t,t_0)P_{pp}B^{\mathrm{T}}(t) + G(t)Q(t)G^{\mathrm{T}}(t)\end{aligned} \tag{3.84}$$

Solving for the cross-covariance, $P_{xp}(t)$, produces

$$P_{xp}(t) = E\left\{\left(\hat{\boldsymbol{x}}(t) - \boldsymbol{x}(t)\right)\left(\hat{\boldsymbol{p}} - \boldsymbol{p}\right)^{\mathrm{T}}\right\} = \Phi(t,t_0)P_{xp}(t_0) + \Psi(t,t_0)P_{pp} \tag{3.85}$$

Taking its derivative gives

$$\dot{P}_{xp}(t) = \dot{\Phi}(t,t_0)P_{xp}(t_0) + \dot{\Psi}(t,t_0)P_{pp} \tag{3.86}$$

Using the derivative definitions and collecting terms as before

$$\dot{P}_{xp}(t) = F(t)\left(\Phi(t,t_0)P_{xp}(t_0) + \Psi(t,t_0)P_{pp}\right) + B(t)P_{pp} \tag{3.87}$$

Substituting in the definition of $P_{xp}(t)$ yields

$$\dot{P}_{xp}(t) = F(t)P_{xp}(t) + B(t)P_{pp} \tag{3.88}$$

The equation for $P_{xp}(t)$ can be also plugged into Eq. (3.84) for even further simplification

$$\dot{P}_{xx}(t) = F(t)P_{xx}(t)+P_{xx}(t)F^{\mathrm{T}}(t)+P_{xp}(t)B^{\mathrm{T}}(t)+B(t)P_{px}(t)+G(t)Q(t)G^{\mathrm{T}}(t) \tag{3.89}$$

where once again the fact that $P_{px}(t) = P_{xp}^{\mathrm{T}}(t)$ is used. The continuous-discrete MVCKF is summarized in Table 8. Since the stability of the continuous-discrete MVCKF is strongly dependent on the sample interval, a general closed-form solution is difficult to obtain and will not be described here.

4. Batch Estimation

In Example II.2, it was shown that the dynamic model could be incorporated in the measurement model to provide a means of estimating the initial states. In that example, however, it was assumed that the dynamic parameters were perfectly known. In this section, the analysis provided in Example II.2 is expanded to include uncertainty in linear dynamic parameters.

The measurement equation to find the initial values for an augmented discrete system is given by

$$\tilde{\boldsymbol{y}}_k = H_{z_k}\Theta(t_k, t_0)\boldsymbol{z}_0 + \boldsymbol{v}_k, \qquad \boldsymbol{v}_k \sim \mathrm{N}(0, R_k) \tag{3.90}$$

Table 8: Continuous-discrete Minimum Variance Consider Kalman Filter

Model	$\dot{\boldsymbol{x}}(t) = F(t)\boldsymbol{x}(t) + B(t)\boldsymbol{p} + G(t)\boldsymbol{w}(t), \quad \boldsymbol{w} \sim \mathrm{N}(0, Q(t))$ $\tilde{\boldsymbol{y}}_k = H_{x_k}\boldsymbol{x}_k + H_{p_k}\boldsymbol{p} + \boldsymbol{v}_k, \quad \boldsymbol{v}_k \sim \mathrm{N}(0, R_k)$
Gain	$K = \left(P_{xx}^- H_x^{\mathrm{T}} + P_{xp}^- H_p^{\mathrm{T}}\right)$ $\times \left(H_x P_{xx}^- H_x^{\mathrm{T}} + H_x P_{xp}^- H_p^{\mathrm{T}} + H_p P_{px}^- H_x^{\mathrm{T}} + H_p P_{pp} H_p^{\mathrm{T}} + R\right)^{-1}$
Update	$\hat{\boldsymbol{x}}^+ = \hat{\boldsymbol{x}}^- + K\left(\tilde{\boldsymbol{y}} - H_x\hat{\boldsymbol{x}}^- - H_p\hat{\boldsymbol{p}}\right)$ $P_{xx}^+ = \left(I - KH_x\right)P_{xx}^- - KH_pP_{px}^-$ $P_{xp}^+ = \left(I - KH_x\right)P_{xp}^- - KH_pP_{pp}$
Propagate	$\dot{\hat{\boldsymbol{x}}}(t) = F(t)\hat{\boldsymbol{x}}(t) + B(t)\hat{\boldsymbol{p}}$ $\dot{P}_{xx}(t) = F(t)P_{xx}(t) + P_{xx}(t)F^{\mathrm{T}}(t) + P_{xp}(t)B^{\mathrm{T}}(t)$ $+B(t)P_{px}(t) + G(t)Q(t)G^{\mathrm{T}}(t)$ $\dot{P}_{xp}(t) = F(t)P_{xp}(t) + B(t)P_{pp}$

Again, for batch estimation it is assumed q sets of measurements are collected and Eqs. (2.2) and (2.3) are still true except that

$$H_z = \begin{bmatrix} H_{z_1}\Theta(t_1,t_0) \\ H_{z_2}\Theta(t_2,t_0) \\ \vdots \\ H_{z_q}\Theta(t_q,t_0) \end{bmatrix} \tag{3.91}$$

Using Eq. (3.14), the augmented state vector can be separated into its components as

$$H_z = \begin{bmatrix} H_{x_1}\Phi(t_1,t_0) & H_{x_1}\Psi(t_1,t_0)+H_{p_1} \\ H_{x_2}\Phi(t_2,t_0) & H_{x_2}\Psi(t_2,t_0)+H_{p_2} \\ \vdots & \vdots \\ H_{x_q}\Phi(t_q,t_0) & H_{x_q}\Psi(t_q,t_0)+H_{p_q} \end{bmatrix} \tag{3.92}$$

or

$$H_x = \begin{bmatrix} H_{x_1}\Phi(t_1,t_0) \\ H_{x_2}\Phi(t_2,t_0) \\ \vdots \\ H_{x_q}\Phi(t_q,t_0) \end{bmatrix}, \quad H_p = \begin{bmatrix} H_{x_1}\Psi(t_1,t_0)+H_{p_1} \\ H_{x_2}\Psi(t_2,t_0)+H_{p_2} \\ \vdots \\ H_{x_q}\Psi(t_q,t_0)+H_{p_q} \end{bmatrix} \tag{3.93}$$

With these definitions, the minimum variance batch methods presented in Section II.2, such as Eqs. (2.30) - (2.32), can be used to estimate the initial values of states given uncertain parameters in both the measurement and dynamic models. Additionally, the block structure of the measurement equation can be taken advantage of to further reduce the computational load. In Eq. (2.31), one of the necessary terms is the value of $H_x^{\mathrm{T}}R^{-1}H_x$, but using the block structure this can be rewritten as

$$H_x^{\mathrm{T}}R^{-1}H_x = \sum_{i=1}^{q} \Phi^{\mathrm{T}}(t_i,t_0)H_{x_i}^{\mathrm{T}}R_i^{-1}H_{x_i}\Phi(t_i,t_0) \tag{3.94}$$

Similarly, the other terms can be summed as

$$H_x^{\mathrm{T}} R^{-1} H_p = \sum_{i=1}^{q} \Phi^{\mathrm{T}}(t_i, t_0) H_{x_i}^{\mathrm{T}} R_i^{-1} \left(H_{x_i} \Psi(t_i, t_0) + H_{p_i} \right) \tag{3.95}$$

$$H_p^{\mathrm{T}} R^{-1} H_x = \sum_{i=1}^{q} \left(H_{x_i} \Psi(t_i, t_0) + H_{p_i} \right)^{\mathrm{T}} R_i^{-1} H_{x_i} \Phi(t_i, t_0) = \left(H_x^{\mathrm{T}} R^{-1} H_p \right)^{\mathrm{T}} \tag{3.96}$$

$$H_p^{\mathrm{T}} R^{-1} H_p = \sum_{i=1}^{q} \left(H_{x_i} \Psi(t_i, t_0) + H_{p_i} \right)^{\mathrm{T}} R_i^{-1} \left(H_{x_i} \Psi(t_i, t_0) + H_{p_i} \right) \tag{3.97}$$

$$H_x^{\mathrm{T}} R^{-1} = \sum_{i=1}^{q} \Phi^{\mathrm{T}}(t_i, t_0) H_{x_i}^{\mathrm{T}} R_i^{-1} \tag{3.98}$$

$$H_p^{\mathrm{T}} R^{-1} = \sum_{i=1}^{q} \left(H_{x_i} \Psi(t_i, t_0) + H_{p_i} \right)^{\mathrm{T}} R_i^{-1} \tag{3.99}$$

Then, Eq. (2.30) in conjunction with Eqs. (3.94) - (3.99) can be used in a batch algorithm, like the one described by Tapley *et al.* [8].

5. Nonlinear Systems

While a linear analysis is important for theoretical purposes, most real-world systems are in fact nonlinear. The Extended Kalman Filter (EKF) provides a quick and easy solution to this by linearizing the nonlinear system about the optimal state estimate. Nonlinear systems can also be used in a consider Kalman filter using the EKF approach [19].

The nonlinear continuous-discrete dynamical system with process noise is defined as

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(t, \boldsymbol{x}, \boldsymbol{p}) + G(t)\boldsymbol{w}(t), \qquad \boldsymbol{w}(t) \sim \mathrm{N}(0, Q(t)) \tag{3.100}$$

$$\tilde{\boldsymbol{y}}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{p}) + \boldsymbol{v}_k, \qquad \boldsymbol{v}_k \sim \mathrm{N}(0, R_k) \tag{3.101}$$

The nonlinear dynamic function, $\boldsymbol{f}(t, \boldsymbol{x}, \boldsymbol{p})$, can be expanded around a reference trajectory, $\boldsymbol{x}^*(t)$, and reference parameter values, $\boldsymbol{p}^*$, using a first-order Taylor series.

$$\boldsymbol{f}(t, \boldsymbol{x}, \boldsymbol{p}) = \boldsymbol{f}(t, \boldsymbol{x}^*, \boldsymbol{p}^*) + \left. \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} \right|_{\boldsymbol{x}^*} (\boldsymbol{x}(t) - \boldsymbol{x}^*(t)) + \left. \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}} \right|_{\boldsymbol{p}^*} (\boldsymbol{p} - \boldsymbol{p}^*) \tag{3.102}$$

Since the reference trajectory is assumed to be free of noise, $\dot{\boldsymbol{x}}^*(t) = \boldsymbol{f}(t, \boldsymbol{x}^*, \boldsymbol{p}^*)$, Eq. (3.100) can be rewritten as

$$\boldsymbol{\delta}\dot{\boldsymbol{x}}(t) = F(t)\boldsymbol{\delta}\boldsymbol{x}(t) + B(t)\boldsymbol{\delta}\boldsymbol{p} + G(t)\boldsymbol{w}(t) \tag{3.103}$$

where

$$F(t) = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}^*} \qquad B(t) = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}}\right|_{\boldsymbol{p}^*} \tag{3.104}$$

$$\boldsymbol{\delta}\dot{\boldsymbol{x}}(t) = \dot{\boldsymbol{x}}(t) - \dot{\boldsymbol{x}}^*(t) \qquad \boldsymbol{\delta}\boldsymbol{x}(t) = \boldsymbol{x}(t) - \boldsymbol{x}^*(t) \qquad \boldsymbol{\delta}\boldsymbol{p} = \boldsymbol{p} - \boldsymbol{p}^* \tag{3.105}$$

This formulation provides a means of estimating the deviations from the reference trajectory as opposed to just the estimated trajectory itself. As mentioned before, in EKFs the current state estimate is used as the reference trajectory, $\boldsymbol{x}^* = \hat{\boldsymbol{x}}$. If the filter was linearized about a reference trajectory then it would be known as a linearized Kalman filter instead of an EKF [8,13]. Note that Eq. (3.103) now has the same form as that of the linear continuous cases (Eq. (3.73)), but only the perturbations are used instead of the actual values.

Similarly, the nonlinear measurement equation (Eq. (3.101)) can be linearized to

$$\boldsymbol{\delta}\tilde{\boldsymbol{y}}_k = H_x\boldsymbol{\delta}\boldsymbol{x}_k + H_p\boldsymbol{\delta}\boldsymbol{p}_k + \boldsymbol{v}_k \tag{3.106}$$

where

$$H_x = \left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_k} \qquad H_p = \left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{p}}\right|_{\hat{\boldsymbol{p}}} \tag{3.107}$$

Again, the above equation now has the linear form used in the above derivations (Eq. (3.1)).

B. Dynamic Estimation Examples

Asteroid orbit and rendezvous has become a great topic of interest in recent years following the success of the Near-Earth Asteroid Rendezvous (NEAR) mission [31],

and because the asteroid known as Apophis coming within striking distance of Earth in the next few decades. To explore this topic from an estimation perspective, two examples are given demonstrating the capabilities and limitations of the MVCKF described above. A completely linear example is presented first to demonstrate the filter's basic properties, followed by a nonlinear example to observe the potential use of the MVCKF in an asteroid rendezvous scenario.

The traditional Kalman filter (TKF) and the augmented Kalman filter (AKF) are also used in the examples to serve as a base for comparison. A derivation of the TKF can be found in a large number of estimation texts including a few references cited in this dissertation [8, 13, 17, 19, 32]. The TKF including process noise is summarized in Table 9. The AKF is the same as the TKF only the augmented state vector, $\boldsymbol{z}$, is used instead of the traditional state vector, $\boldsymbol{x}$.

The dwarf planet Ceres is used as the rendezvous object, because of its roughly spherical nature and that its gravity needs to be accounted for in addition to its angular spin rate. Table 10 contains a list of a few of the current estimates for the physical parameters of the large asteroid.

In both of the examples, process noise is ignored to more effectively compare the filters with minimal tuning. In addition, all filters are given the same initial estimates and covariance values to judge how they respond to the same set of data.

1. Example III.1 - Freefall

In this first example, the wholly unrealistic assumption of constant gravity is made to observe how each of the three filters handles linear parameter error introduced into the dynamic model. Using this assumption, the system dynamics are defined as

$$\ddot{x} = 0 \qquad \ddot{y} = -g \tag{3.108}$$

Table 9: Discrete Traditional Kalman Filter

Model	$\boldsymbol{x}_{k+1} = \Phi_k \boldsymbol{x}_k + \Psi_k \boldsymbol{p} + \Gamma_k \boldsymbol{w}_k, \qquad \boldsymbol{w}_k \sim \mathrm{N}(0, Q_k)$ $\tilde{\boldsymbol{y}}_k = H_{x_k} \boldsymbol{x}_k + \boldsymbol{v}_k, \qquad \boldsymbol{v}_k \sim \mathrm{N}(0, R_k)$
Gain	$K = P_{xx}^- H_x^{\mathrm{T}} \left(H_x P_{xx}^- H_x^{\mathrm{T}} + R \right)^{-1}$
Update	$\hat{\boldsymbol{x}}^+ = \hat{\boldsymbol{x}}^- + K \left(\tilde{\boldsymbol{y}} - H_x \hat{\boldsymbol{x}}^- \right)$ $P_{xx}^+ = \left(I - K H_x \right) P_{xx}^-$
Propagation	$\hat{\boldsymbol{x}}_{k+1}^- = \Phi_k \hat{\boldsymbol{x}}_k^+ + \Psi_k \boldsymbol{p}$ $P_{xx,k+1}^- = \Phi_k P_{xx,k}^+ \Phi_k^{\mathrm{T}} + \Gamma_k Q_k \Gamma_k^{\mathrm{T}}$

Table 10: Physical Parameters of the Dwarf Planet Ceres [33–35]

Equatorial Radius (km)	487.3 ± 1.8
Polar Radius (km)	454.7 ± 1.6
Mass (kg)	9.43e20 ± 0.07e20
Sidereal Rotation Period (h)	9.074170 ± 2e-6

where x is the lateral position to the center of Ceres, y is the radial position, and g is the constant gravity of the asteroid, which is about 0.265 m/s^2 at the surface. Thus, the state-space formulation for the TKF and MVCKF uses the following for its dynamic model matrix values

$$F(t) = F = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \qquad B(t) = B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -1 \end{bmatrix} \tag{3.109}$$

For the AKF, the values are

$$F_z(t) = F_z = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.110}$$

since $\dot{g} = 0$.

Measurements are made using both position values which forces the measurement model matrix values to be

$$H_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \qquad H_p = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{3.111}$$

for the TKF and MVCKF. For the AKF

$$H_z = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{3.112}$$

Truth and initial estimate values used in the simulations are shown in Table 11. Note that the large discrepancy between the true and estimated gravity constants was

selected to exaggerate each filter's performance. The diagonals of the initial state and parameter covariances, P_{xx_0} and P_{pp}, were set to the square of the difference between the true and estimated values. This allowed the initial error to be well within the initial 3σ bound for each state. All initial covariances off of the diagonal were set to zero. The initial cross-covariance between the states and parameters, P_{xp_0}, was also set to zero. The measurement standard deviation was set to 2 km for both measurements, and measurements were collected at a sampling rate of 10 Hz.

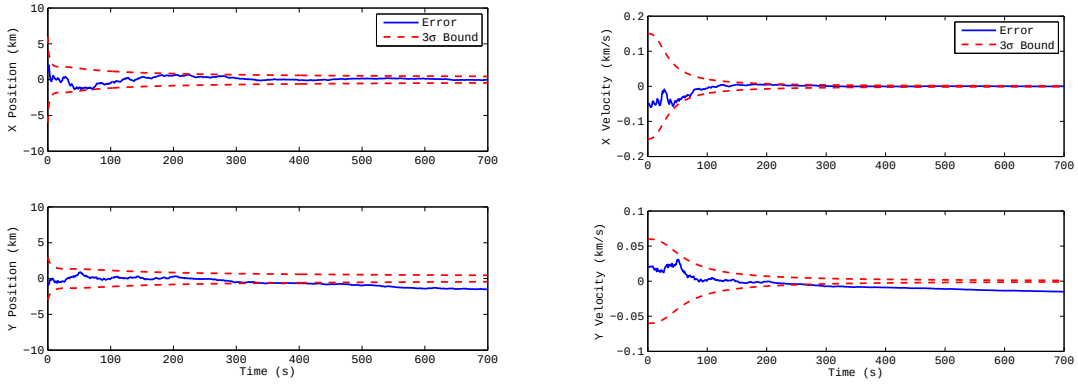
The results of a single simulation are shown in Figures 11 - 13. The estimation results for g from the AKF are shown in Figure 14. Since the gravity constant only acts in the Y direction, the X direction is completely unaffected in all three filters by the error in the parameter. The TKF has no way to account for the incorrect parameter, thus, the error in both position and velocity increase with time and the covariance is unable to capture it. The AKF does an excellent job of tracking position and velocity in both directions. Also, because all of the states and the parameter are directly observable the AKF manages to minimize the error in the gravity constant as well.

The MVCKF has similar state performance to that of the AKF and also has stable bounds around the error. Figure 15 shows the difference between the AKF and MVCKF state estimates. The disparity between the two filters results from the fact that the AKF state covariance decreases with time as the parameter estimate

Table 11: State and Parameter Values Used in Example III.1

	x_0 (km)	$\dot{x}_0$ (km/s)	y_0 (km)	$\dot{y}_0$ (km/s)	g (m/s^2)
Truth	0	0.3	50	0	0.220
Estimated	2	0.25	49	0.02	0.265

improves, while the covariance for the MVCKF states reaches a steady value. This is because the MVCKF parameter covariance cannot improve beyond a certain point since it is limited by the constant parameter covariance. Despite this limitation, this simple example shows that the MVCKF is capable of effectively estimating the states in the presence of parameter uncertainty. Moreover, the MVCKF requires fewer computations than the AKF since the calculations required are a subset of the fully augmented system. This computation reduction further improves as the number of considered parameters increases.

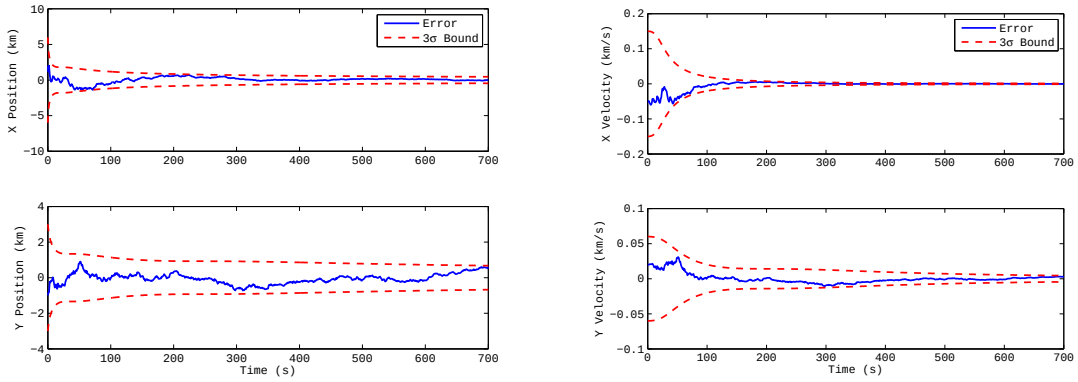


(a) Position (b) Velocity

Fig. 11: State Errors with 3σ Covariance Bounds for the TKF in Example III.1

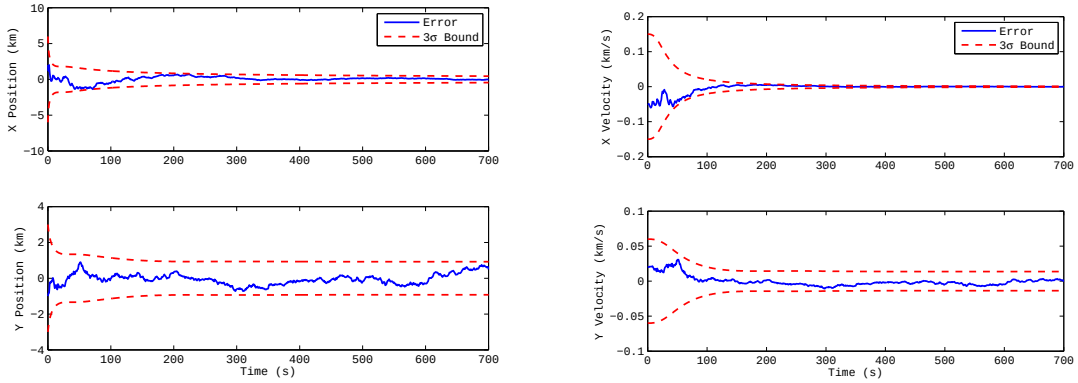
2. Example III.2 - Asteroid Rendezvous

While the previous example was useful in showing some of the characteristics of the different filters used to handle parameter errors, the assumptions used cannot be applied to a real world application. In this example some of those assumptions are relaxed. Specifically, the gravity constant must be allowed to vary with radial distance and the angular spin rate must be taken into account since the spacecraft



(a) Position (b) Velocity

Fig. 12: State Errors with 3σ Covariance Bounds for the AKF in Example III.1



(a) Position (b) Velocity

Fig. 13: State Errors with 3σ Covariance Bounds for the MVCKF in Example III.1

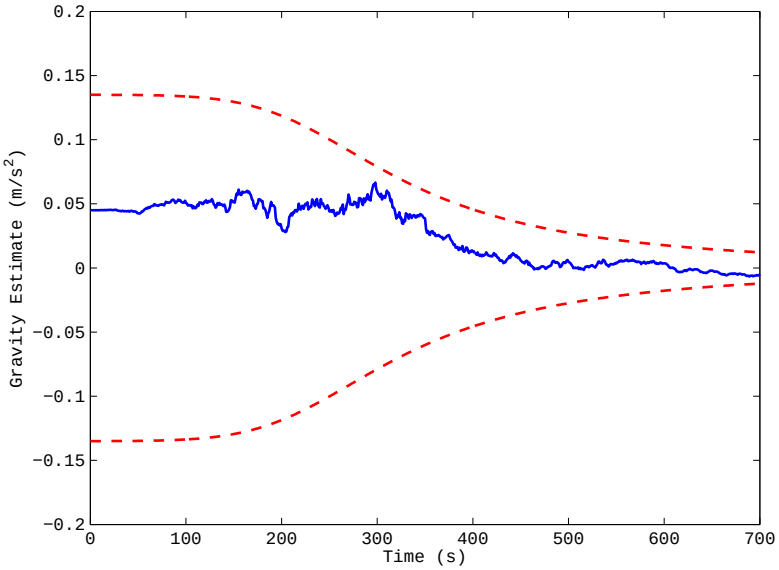


Fig. 14: Gravity Constant Error in the AKF for Example III.1

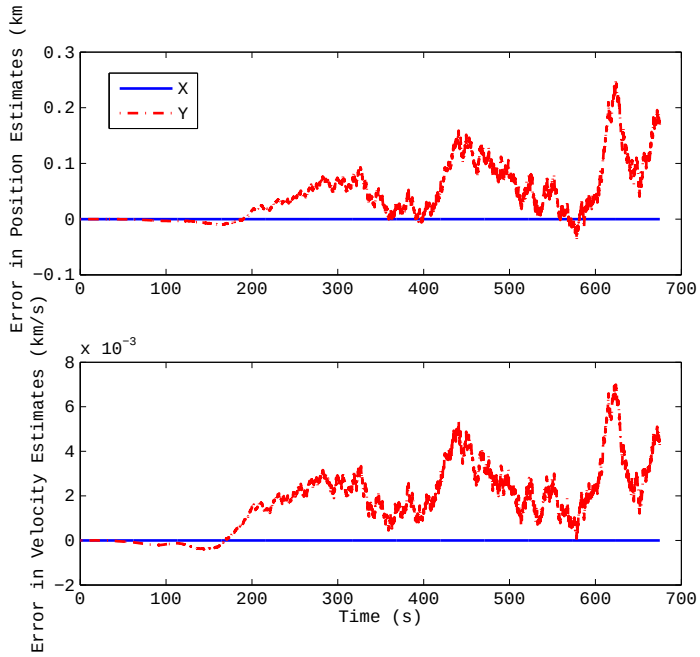


Fig. 15: Difference Between AKF and MVCKF State Estimates in Example III.1

is to land at a desired location. By applying a body-fixed frame to Ceres, where the Z-axis is aligned with the rotational axis, the dynamics of a landing body are known to be [36]

$$\ddot{x} = 2\omega\dot{y} + \omega^2 x - \frac{\mu}{r^3}x \tag{3.113}$$

$$\ddot{y} = -2\omega\dot{x} + \omega^2 y - \frac{\mu}{r^3}y \tag{3.114}$$

where ω is the angular spin rate, μ is the gravitational coefficient for Ceres, and r is the radial distance from the center of Ceres or $r = \sqrt{x^2 + y^2}$. Both constant parameters, μ and ω, appear nonlinearly in the dynamic equations and an Extended form of each filter must be employed. The four states to be estimated are both positions and velocities as before, but the parameter vector is now defined as $\boldsymbol{p} = [\mu \quad \omega]^{\mathrm{T}}$. Using the linearization method shown in Section 5, the dynamic model matrices for the TKF and MVCKF are found to be

$$F(t) = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \omega^2 - \frac{\mu}{r^3} + \frac{3\mu x^2}{r^5} & \frac{3\mu xy}{r^5} & 0 & 2\omega \\ \frac{3\mu xy}{r^5} & \omega^2 - \frac{\mu}{r^3} + \frac{3\mu y^2}{r^5} & -2\omega & 0 \end{bmatrix} \tag{3.115}$$

$$B(t) = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{p}} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ -\frac{x}{r^3} & 2\left(\dot{y} + \omega x\right) \\ -\frac{y}{r^3} & 2\left(-\dot{x} + \omega y\right) \end{bmatrix} \tag{3.116}$$

For the AKF, the dynamic matrix is

$$F_z(t) = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{z}} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \omega^2 - \frac{\mu}{r^3} + \frac{3\mu x^2}{r^5} & \frac{3\mu xy}{r^5} & 0 & 2\omega & -\frac{x}{r^3} & 2\left(\dot{y} + \omega x\right) \\ \frac{3\mu xy}{r^5} & \omega^2 - \frac{\mu}{r^3} + \frac{3\mu y^2}{r^5} & -2\omega & 0 & -\frac{y}{r^3} & 2\left(-\dot{x} + \omega y\right) \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.117}$$

To provide ample observability to all the filters, measurements of both position and velocity in the body frame are assumed to be available at each time step. This forces the measurement model matrices to be

$$H_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad H_p = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{3.118}$$

for the TKF and MVCKF. For the AKF

$$H_z = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{3.119}$$

The estimated values used to initialize the simulations below are shown in Tables 12 and 13. This table also includes the standard deviations used during the Monte Carlo simulations to define the location of the truth. The estimated values of the parameters and their associated standard deviations are based on the values and

tolerances given in Table 10. The initial variance for each state and parameter is the square of the given standard deviation. Off-diagonals of the inital covariance matrices are set to zero as well as the initial cross-covariance, P_{xp_0}, as before. Measurements were sampled at a rate of 1 Hz with the position and velocity standard deviations being 10 km and 0.01 km/s, respectively.

In order to assess how sensitive the system was to noise due to the linearized model, consistency testing via Monte Carlo simulations was performed for all three filters. Results from a single Monte Carlo run are presented first followed by the consistency test results.

3. Single Run Results

Figures 16 - 18 contains the results from the final run of the Monte Carlo simulations. The TKF appears to perform adequately for both position estimates save for the Y direction's slight divergence at the end. The velocity estimates, however, are well outside of their covariance bounds. Both the AKF and MVCKF effectively calculate both position and velocity estimates. The MVCKF has slightly larger covariance bounds, similar to Example III.1, since the AKF is still able to estimate the gravitational parameter as shown in Figure 19. The AKF was unable to improve the angular spin rate estimate because of numerical difficulties resulting from its high precision. Thus, even with nonlinearities all three estimators produce results similar to that of Example III.1.

4. Consistency Testing

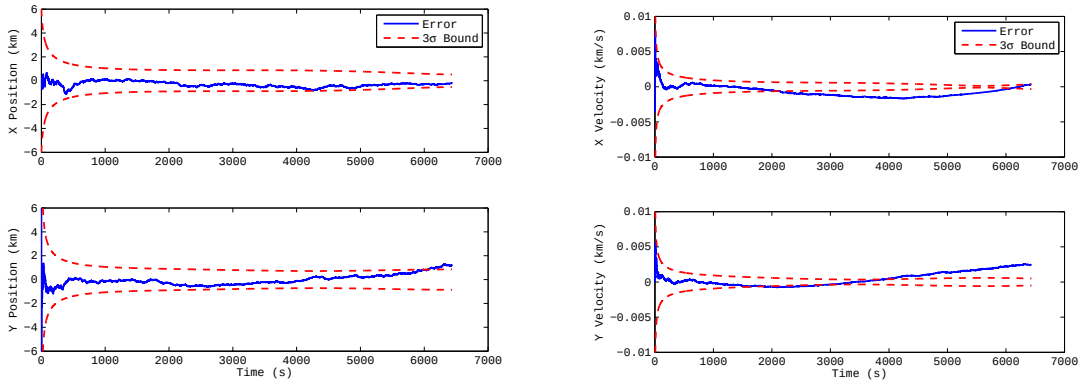
After implementing a Kalman filter for a particular application, a series of tests can be performed to ensure that it is providing the optimal results. These tests can be used to confirm certain assumptions in the dynamic or measurement models, as

Table 12: State Estimated Values and Standard Deviations used in Example III.2

States	x_0 (km)	$\dot{x}_0$ (km/s)	y_0 (km)	$\dot{y}_0$ (km/s)
Estimated Value	2	0.410	835	-0.01
Standard Deviation	2	0.015	15	0.01

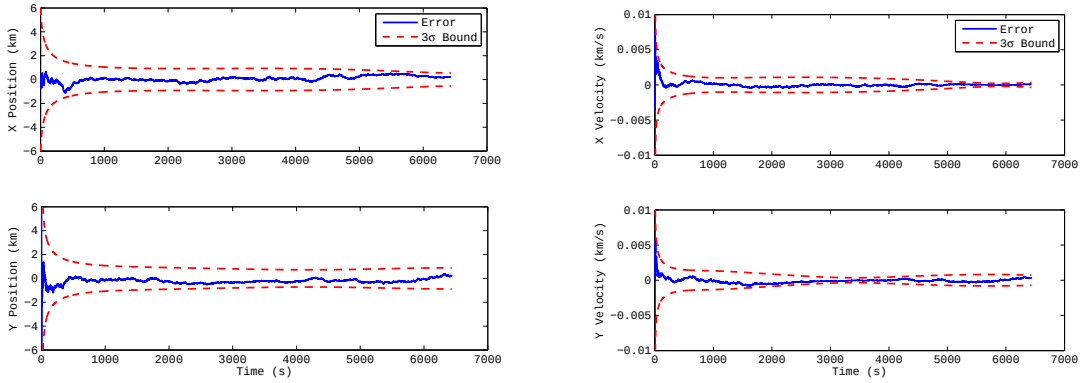
Table 13: Parameter Estimated Values and Standard Deviations Used in Example III.2

Parameters	μ (km^3/s^2)	ω (rad/s)
Estimated Value	63.57	1.923406e-4
Standard Deviation	0.63	2e-10



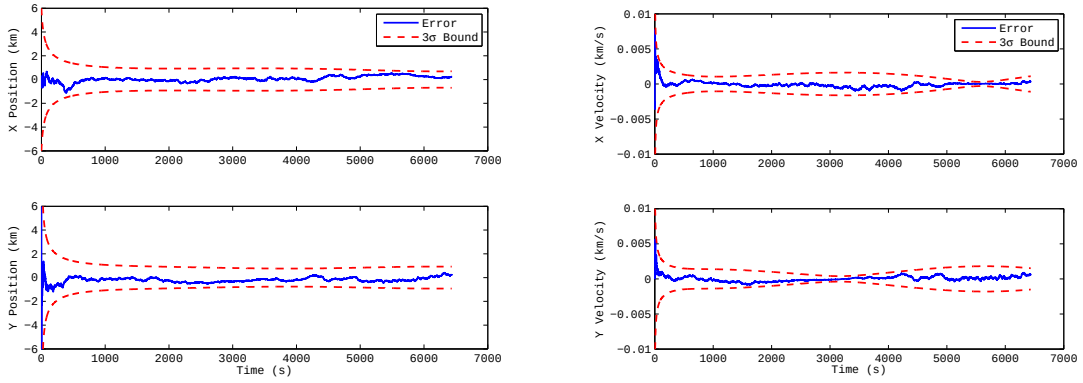
(a) Position (b) Velocity

Fig. 16: State Errors with 3σ Covariance Bounds for the TKF in Example III.2 - Single Run



(a) Position (b) Velocity

Fig. 17: State Errors with 3σ Covariance Bounds for the AKF in Example III.2 - Single Run



(a) Position (b) Velocity

Fig. 18: State Errors with 3σ Covariance Bounds for the MVCKF in Example III.2 - Single Run

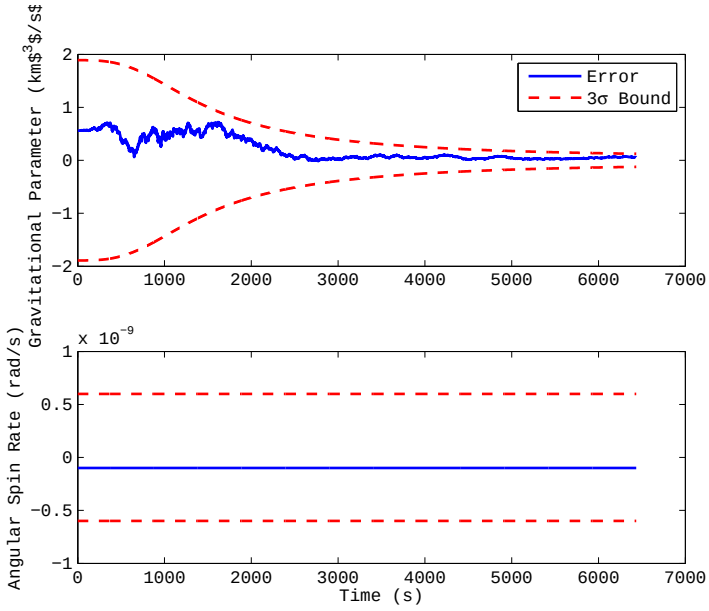


Fig. 19: Gravitational Constant and Angular Spin Rate Error in AKF for Example III.2 - Single Run

well as validate the state estimates being produced [13,32]. Moreover, the *consistency* of the filter can be verified to ensure that optimality will continue as time progresses. While a basic understanding of statistics is assumed, additional explanations of the statistic terminology used can be found in Stark and Woods' textbook [37].

One test is the Normalized Error Square (NES) test for state estimates. This test simultaneously can determine if the state estimate errors have zero mean and if they have a covariance equal to P_{xx}. Defining the state error as

$$\tilde{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_k - \boldsymbol{x}_k \tag{3.120}$$

where k is the value of the state at time t_k. The NES test statistic can then be defined as

$$\epsilon_k = \tilde{\boldsymbol{x}}_k^{\mathrm{T}} P_{xx,k}^{-1} \tilde{\boldsymbol{x}}_k \tag{3.121}$$

Under the hypothesis that the state errors have a Gaussian distribution, ϵ_k will have a chi-squared distribution with degrees of freedom equal to the number of states, n,

or

$$E\left\{\epsilon_k\right\} = n \tag{3.122}$$

Frequently, Monte Carlo simulations are used to provide N independent samples of the test statistics. An averaged NES statistic can then be employed and is defined as

$$\bar{\epsilon}_k = \frac{1}{N}\sum_{i=1}^{N}\epsilon_k(i) \tag{3.123}$$

where $\epsilon_k(i)$ can be interpreted as the test statistic of the i^{th} simulation at time t_k. Thus, $N\bar{\epsilon}_k$ will have a chi-squared distribution with Nn degrees of freedom. The hypothesis that *the state estimation errors are consistent with the filter calculated covariances* can be evaluated using a chi-squared test and is accepted if

$$\bar{\epsilon}_k \in [\zeta_1, \zeta_2] \tag{3.124}$$

where ζ_1 and ζ_2 are evaluated based on the tail probabilities of the chi-squared density function for a two-sided probability region. It is common to use a 95% confidence interval which is specified using $100\,(1-\alpha)$ where $\alpha = 0.05$ in this case. As an example, let $n = 2$ and $N = 50$ then $\chi^2_{nN}(0.975) = 74.22$ and $\chi^2_{nN}(0.025) = 129.56$. This generates the acceptance range of $\zeta_1 = \chi^2_{nN}(0.975)/N = 1.48$ and $\zeta_2 = \chi^2_{nN}(0.025)/N = 2.59$.

This test can also be used for a single test, $N = 1$. In this case, $\zeta_1 = \chi^2_{nN}(0.975) = 0.05$ and $\zeta_2 = \chi^2_{nN}(0.025) = 7.38$. Note how much smaller the range is for this scenario than when $N = 50$. This demonstrates how the variability of the test decreases as the number of repeated simulations increases.

When using the NES test, if a bias exists in the state estimates then the estimation error in Eq. (3.121) will increase. Additionally, if the filter becomes over-confident in its estimates, P_{xx}^{-1} will become large which again increases the value of ϵ_k. If the

NES test fails, then a separate bias test can be performed to identify the source of the problem. The test statistic for the Normalized Mean Error (NME) test is given by

$$[\bar{\mu}_k]_j = \frac{1}{N}\sum_{i=1}^{N}\frac{[\tilde{\boldsymbol{x}}_k(i)]_j}{\sqrt{[P_{xx,k}(i)]_{jj}}} \tag{3.125}$$

where $[\tilde{\boldsymbol{x}}_k(i)]_j$ is interpreted as the j^{th} component of the estimated state error at time t_k. Under ideal conditions, $[\mu_k]_j$ will have a a distribution given by N(0,1) which means the averaged statistic, $[\bar{\mu}_k]_j$, is distributed by N(0,1/N). Then the hypothesis that *the mean of the j^{th} component of the state estimation error has zero mean* is accepted if

$$[\bar{\mu}_k]_j \in \left[-\frac{r}{\sqrt{N}}, \frac{r}{\sqrt{N}}\right] \tag{3.126}$$

where r is the tail probabilities of the normalized Gaussian distribution for a given confidence interval. For instance, a two-sided 95% confidence interval gives r a value of 1.96.

In addition to evaluating the consistency of the state estimation errors, the NES and NME tests can also be used with the residual errors, $\boldsymbol{e} = \tilde{\boldsymbol{y}} - \hat{\boldsymbol{y}}$. The covariance of the residual errors is defined as

$$P_y = E\left\{\boldsymbol{e}\boldsymbol{e}^{\mathrm{T}}\right\} \tag{3.127}$$

An alternative covariance can also be used, where it is assumed that there are no errors in the state estimates. This reduces the residual error covariance to be equivalent to the measurement covariance, R, for the traditional Kalman filter. For a consider Kalman filter the more restrictive residual error covariance is

$$P_y = H_p P_{pp} H_p^{\mathrm{T}} + R \tag{3.128}$$

The whiteness test can also be applied to the residuals to determine if they are uncorrelated and can be truly treated as white noise. The sample autocorrelation

statistic for the full residual vector is

$$\bar{\rho}_k(j) = \frac{1}{\sqrt{m}} \sum_{i=1}^{N} \boldsymbol{e}_k^{\mathrm{T}}(i) \left[\sum_{i=1}^{N} \boldsymbol{e}_k(i) \boldsymbol{e}_k^{\mathrm{T}}(i) \right]^{-1/2} \left[\sum_{i=1}^{N} \boldsymbol{e}_j(i) \boldsymbol{e}_j^{\mathrm{T}}(i) \right]^{-1/2} \boldsymbol{e}_j(i) \tag{3.129}$$

where m is the length of $\boldsymbol{e}$ and $k \neq j$. For large enough N a normal approximation can be made for the density of the whiteness statistic via the central limit theorem and has mean zero with variation $1/N$. Therefore, the residuals can be considered white if

$$\bar{\rho}_k(j) \in \left[-\frac{r}{\sqrt{N}}, \frac{r}{\sqrt{N}} \right] \tag{3.130}$$

which is the same acceptance region based on the normalized Gaussian distribution used for the NME test.

Using the theory outlined above, a set of 500 Monte Carlo runs was performed on the scenario presented in the previous section. These tests were used to highlight estimation biases and the effectiveness of the covariance estimates in accounting for error in the state estimates. The NES test is used to analyze the overall nature of the relationship between the covariance and the state and/or residual error. Figures 20 and 21 show the results of the NES test for both the state and residual error, respectively. While the NES test on the states shows a clear bias in the TKF, the NES test on the residuals shows that no bias exists in the measurement model. Thus, despite the bias in the state error the measurement equations for the TKF can still be treated as Gaussian processes. As a result, no further consistency tests on the residuals will be presented since the state estimate error analysis can be performed directly.

The NES test amplified both the state and covariance estimate error from the TKF. This results in the large values shown in Figure 20 proving that the TKF is unable to account for parameter error in the dynamic model. On the other hand, the low state errors and the effective covariance calculations resulted in acceptable test statistics for the both the AKF and the MVCKF.

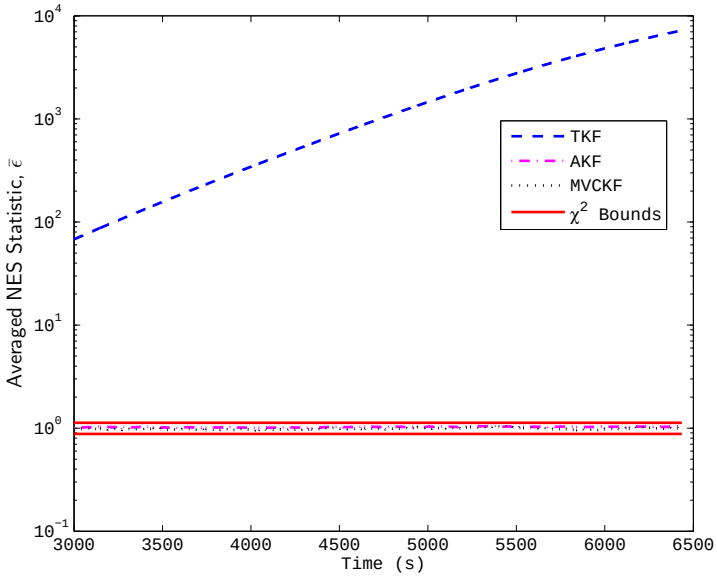


Fig. 20: NES Test Results - State Error

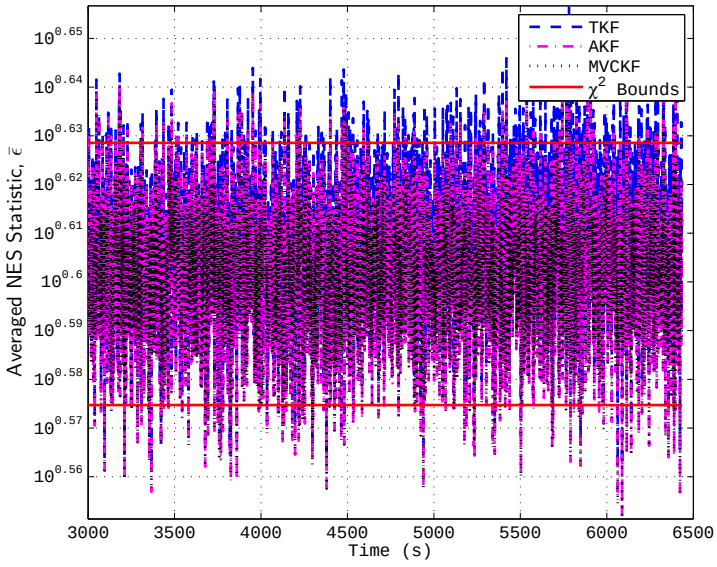


Fig. 21: NES Test Results - Residual Error

The NME test is used to analyze each state with its associated variance to root out the cause of biases that appeared in the NES test results. The NME test results for the states are shown in Figure 22. The AKF and MVCKF results only confirm the results of the NES test, since all of their state test statistics are well below the required threshold. The critical observation, though, is that the MVCKF performed as well as the AKF without direct estimation of the parameters. The other main observation is that the bias in the TKF results are caused by the poor velocity estimates as shown in Figure 16(b). In short, the AKF and MVCKF consistently provide excellent state estimates and the TKF has trouble accounting for the error in the parameters.

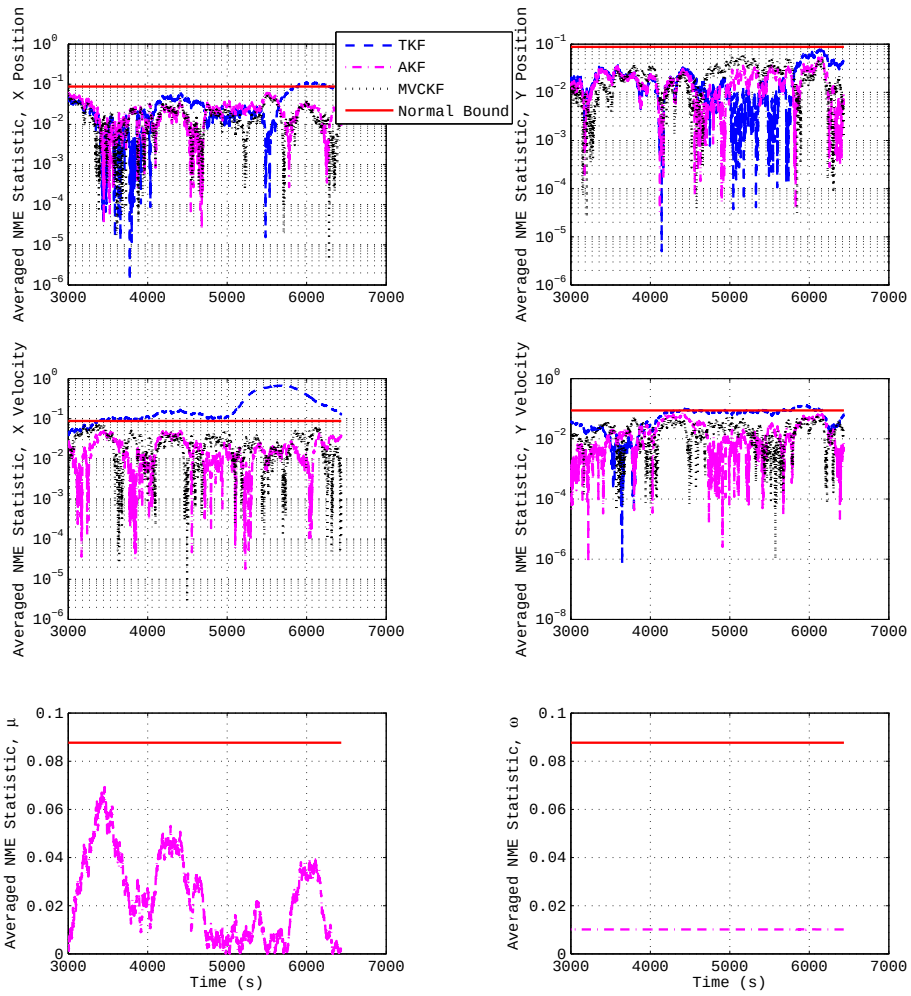


Fig. 22: NME Test Results - State Error

CHAPTER IV

ALTERNATE METHODS TO ACCOUNT FOR PARAMETER UNCERTAINTY

As mentioned in Chapter I, multiple methods have been developed over the years to account for parameter uncertainty without directly estimating the parameters themselves. While the past two chapters have focused on the use of the consider methodology in both static and dynamics systems, this chapter examines these alternative methods from a theoretical standpoint, then uses simple examples to compare their results to that produced by the consider Kalman filter.

A. Process Noise and the CKF

The current popular method of accounting for dynamic parameter errors in filters is to adjust the process noise covariance until acceptable results are achieved. In essence, the idea is that additional uncertainty is introduced into the dynamic model to account for neglected higher-order effects [10]. These concepts can also be applied to uncertainty in the measurement model by artificially increasing the measurement covariance as will be shown in Chapter V. Consider Kalman filtering provides an alternative method to the use of process noise by including the parameter error covariance and its cross-covariance with the states in the propagation equations. Schlee *et al.* [21] performed a comparison of the use of process noise with two different versions of the Minimum Variance Consider Kalman Filter (MVCKF) developed by Schmidt [18]. They found that the sole use of process noise has slightly reduced accuracy, but is capable of accounting for round-off error in simple computer systems when properly tuned. Schlee *et al.* further conclude that there are particular situations where the

use of process noise will outperform the consider Kalman filter because of round-off error. In truth, a consider Kalman filter could be used to accurately account for the parameter error while process noise could also be used to prevent divergence due to computational effects. This section compares the use of process noise in a traditional Kalman filter and the consider Kalman filter theoretically. The theoretical results are then analyzed using a simple freefall example.

If no measurement model parameters are considered, then the update equations are equivalent for the TKF and the MVCKF. Thus, not surprisingly, the differences in the filters are caused by the propagation equations. Given that the estimated dynamic parameters are used instead of the true parameter values, a simple substitution in the truth equation gives a first initial guess as to the optimal value of the process noise. The assumption is made that the estimated parameters vary from the truth via a Gaussian distribution, or

$$\hat{\boldsymbol{p}} = \boldsymbol{p} + \boldsymbol{\beta}, \qquad \boldsymbol{\beta} \sim \mathrm{N}(0, P_{pp}) \tag{4.1}$$

Plugging this result into the true propagation equation results in

$$\boldsymbol{x}_{k+1} = \Phi_k \boldsymbol{x}_k + \Psi_k \boldsymbol{p} + \Psi_k \boldsymbol{\beta} \tag{4.2}$$

Comparing this result to the propagation equation for the traditional Kalman filter with process noise (Table 9) defines the variables as

$$\Gamma_k = \Psi_k, \qquad \boldsymbol{w}_k = \boldsymbol{w} = \boldsymbol{\beta}, \qquad Q_k = Q = P_{pp} \tag{4.3}$$

Alternatively, since the process noise covariance, Q_k, only appears in the state covariance propagation equation, this is the equation that must be the same for the two filters to be equivalent. Equating the covariance propagation equations for the

TKF and MVCKF gives

$$P^{-}_{xx,k+1} = \Phi_k P^{+}_{xx,k} \Phi^{\mathrm{T}}_k + \Gamma_k Q_{\mathrm{TKF},k} \Gamma^{\mathrm{T}}_k$$

$$= \Phi_k P^{+}_{xx,k} \Phi^{\mathrm{T}}_k + \Phi_k P^{+}_{xp,k} \Psi^{\mathrm{T}}_k + \Psi_k P^{+}_{px,k} \Phi^{\mathrm{T}}_k + \Psi_k P_{pp} \Psi^{\mathrm{T}}_k + \Gamma_k Q_{\mathrm{CKF},k} \Gamma^{\mathrm{T}}_k \tag{4.4}$$

or

$$\Gamma_k Q_{\mathrm{TKF},k} \Gamma^{\mathrm{T}}_k = \Phi_k P^{+}_{xp,k} \Psi^{\mathrm{T}}_k + \Psi_k P^{+}_{px,k} \Phi^{\mathrm{T}}_k + \Psi_k P_{pp} \Psi^{\mathrm{T}}_k + \Gamma_k Q_{\mathrm{CKF},k} \Gamma^{\mathrm{T}}_k \tag{4.5}$$

Thus, for the traditional process noise to accurately portray the parameter variance, the complete process noise covariance update, $\Gamma_k Q_k \Gamma^{\mathrm{T}}_k$, is not only a function of the parameter covariance, P_{pp}, but also time-dependent on the cross-covariance, P_{xp}. Except in special cases, neglecting the cross-covariance terms will generate incorrect state covariance estimates. The CKF process noise term, $Q_{\mathrm{CKF},k}$ differs from that of the TKF process noise, because it is accounting for only high-order nonlinearities and computer round-off. Dynamic parameter uncertainties are already accounted for in the CKF by its underlying structure.

1. Example IV.1

The freefall example from Chapter III is used again here to compare the use of process noise and the consider methodology. Again, the system dynamics for this linear example are defined as

$$\ddot{x} = 0 \qquad \ddot{y} = -g \tag{4.6}$$

where x is the lateral position to the center of a rigid body, y is the radial position, and g is the gravity of the body, which will be assumed to be constant. The discrete

state-space formulation for this system is known to be [36]

$$\Phi_k = \Phi(t, t_0) = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad \Psi_k = \Psi(t, t_0) \begin{bmatrix} 0 \\ -0.5\Delta t^2 \\ 0 \\ -\Delta t \end{bmatrix} \tag{4.7}$$

where $\Delta t = t - t_0$ is the time interval between measurements.

Measurements are made using both position values which forces the measurement model matrix values to be

$$H_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \qquad H_p = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{4.8}$$

since only the dynamic parameter, g, is being considered.

Truth and initial estimate values used in the simulations are shown in Table 14. Note that the large discrepancy between the true and estimated gravity constants was selected to exaggerate performance. The diagonals of the initial state and parameter covariances, P_{xx_0} and P_{pp}, were set to the square of the difference between the true and estimated values. This allowed the initial error to be well within the initial 3σ bound for each state. All initial covariances off the diagonal were set to zero. The initial cross-covariance between the states and parameters, P_{xp_0}, was also set to zero. The measurement standard deviation was set to 2 km for both measurements and measurements were collected at a sampling rate of 1 Hz.

Table 14: State and Parameter Values Used in Example IV.1

	x_0 (km)	$\dot{x}_0$ (km/s)	y_0 (km)	$\dot{y}_0$ (km/s)	g (m/s^2)
Truth	0	0.3	50	0	100
Estimated	2	0.25	49	0.02	120

Four separate filter cases are examined:

1. Traditional Kalman Filter with No Process Noise
2. Traditional Kalman Filter with Parameter Error Noise Only
3. Traditional Kalman Filter with Cross-Covariance and Parameter Error Noise
4. Minimum Variance Consider Kalman Filter

The first case assumes $\boldsymbol{w}_k = 0$ for all values of k. The second case assumes the values of the process noise are given by Eq. (4.3). The third adds the additional cross-covariance terms to the process noise as in Eq. (4.5). In this variation, however, the process noise is assumed constant and the steady-state value of P_{xp} from the MVCKF is used for each time step. The final case uses the complete consider filter described in Table 6.

Figures 23 - 26 displays the results of the Y-direction position error and covariance for a sample run. The values on the x and $\dot{x}$ states are not shown as they are unaffected by these results. $\dot{y}$ shows similar results to that for y, but they are less pronounced. Case 1 produces the best covariance estimates, but the state estimates quickly leave their covariance bounds as seen in previous examples. Case 2 observes the next best covariance estimates, but a clear negative bias is present in the results. In this particular scenario, the bias had a mean value of approximately -2 km. While Case 2 is theoretically unbiased, the increased systemic error is a result of the over-confidence in the estimate. By including the cross-covariance terms in Case 3 the variance estimates are more accurate and the bias is reduced. This also verifies that an appropriately selected process noise covariance can simulate the consider Kalman filter. Alternatively, the consider methodology could be used to provide an initial estimate of the process noise components, if the user was forced to use process noise. It

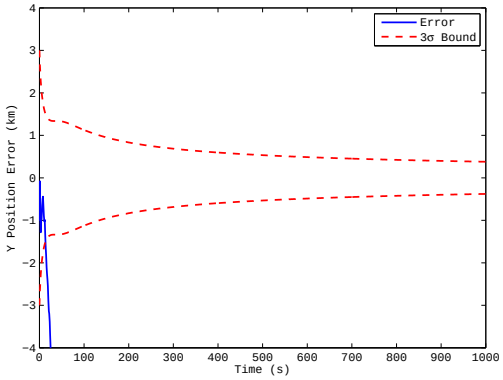
should be noted that the Case 3 covariance bound is slightly larger than the MVCKF covariance bound.

It is also noticeable that the MVCKF has a slight negative bias despite the fact that it is theoretically unbiased. This results from the use of a constant estimated parameter value even though the estimate is presumed to have a Gaussian distribution. This bias can be further reduced by adding process noise to the MVCKF or increasing the parameter covariance at the cost of increasing the state covariance (Figure 27). Using process noise values of 0.1 and 0.05 on y and $\dot{y}$, respectively, reduced the mean bias of the last 900 position estimates from -0.44 km to -0.08; however, the standard deviation increased from 0.89 km to 1.19 km. Similarly, the velocity mean bias decreased from -0.14 km/s to -0.07 km/s by adding process noise, but the standard deviation increased from 0.105 km/s to 0.286 km/s.

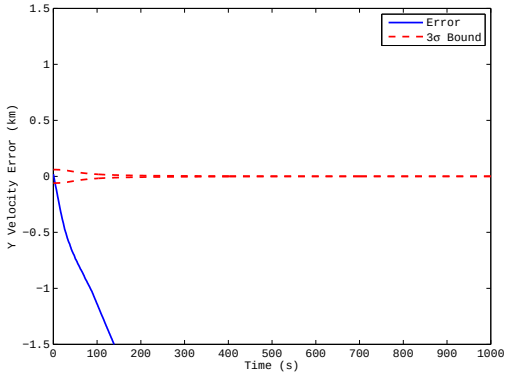
Based on this analysis it is clear that process noise can be used to account for parameter error, but involves a significant tuning process to remove biases as well as accurately portray the error covariance. On the other hand, the MVCKF does an excellent job estimating the covariance bounds, but a slight bias can be produced if a parameter is given a single value instead of being drawn from a distribution. This bias can be reduced by including process noise in the MVCKF dynamic model, but at the cost of giving an inflated covariance estimate.

B. Reduced-order Filters

Other methods were developed to account for parameter uncertainty since many viewed the application of process noise as an *ad hoc* or improvisation method. The consider methodology presented in Chapter III is only a subset of a larger group of filters known as reduced-order filters. Reduced-order filters attempt to reduce compu-

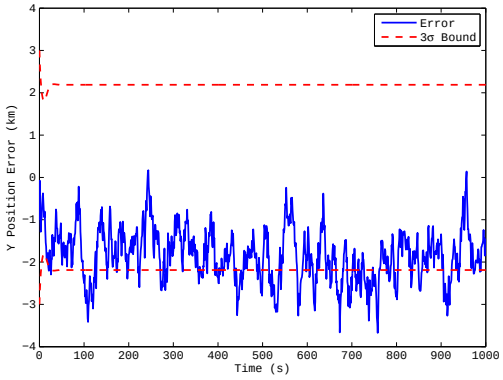


(a) Position

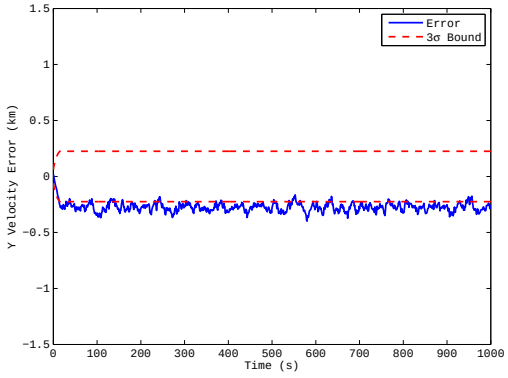


(b) Velocity

Fig. 23: Example IV.1 Results for Case 1

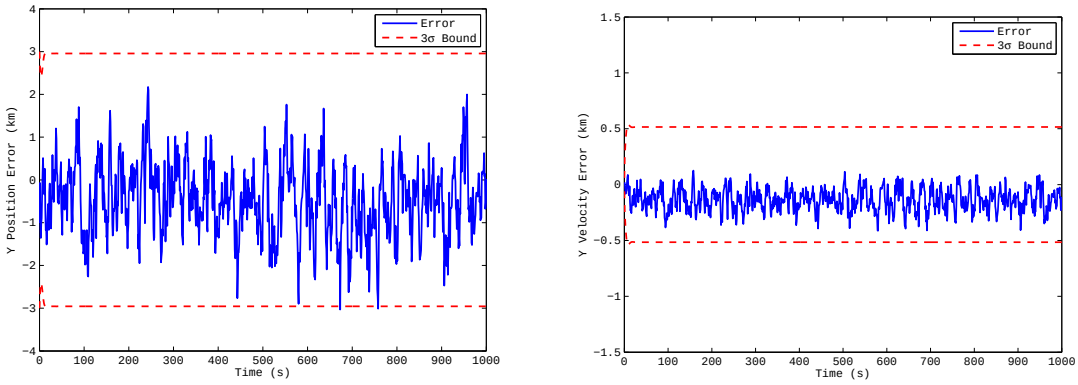


(a) Position



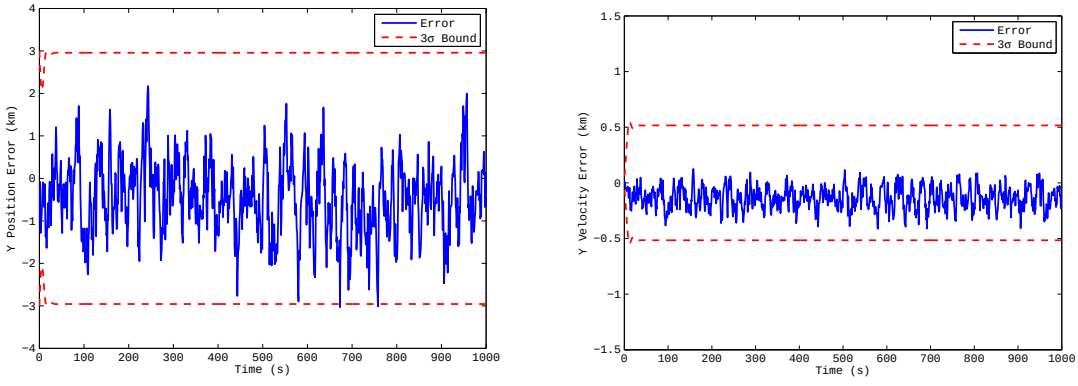
(b) Velocity

Fig. 24: Example IV.1 Results for Case 2



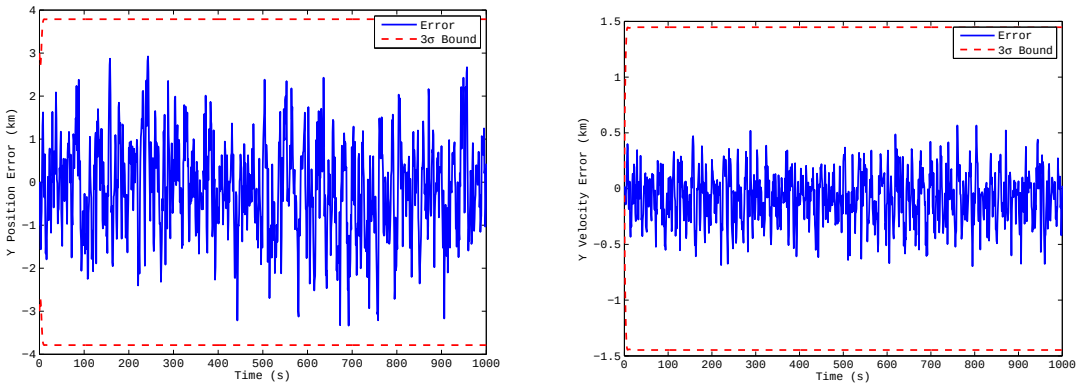
(a) Position (b) Velocity

Fig. 25: Example IV.1 Results for Case 3



(a) Position (b) Velocity

Fig. 26: Example IV.1 Results for Case 4



(a) Position (b) Velocity

Fig. 27: Example IV.1 Results when Process Noise is Added to the MVCKF

tational overhead by lowering the dimensionality of high-order system models while at the same time minimizing performance loss. This section presents two additional reduced-order filters that have been previously developed. The first is an alternate derivation of the consider Kalman filter using an augmented measurement model. Comparisons are made between the two CKFs. Then a completely different filter is presented which directly reduces the dimensionality by multiplying the augmented state vector, $\boldsymbol{z}$, by a transformation matrix.

1. Alternate Consider Kalman Filter Derivation

Tapley *et al.* provide an alternate method to derive a Kalman filter based on the consider methodology [8]. Their approach described below provides the basis for the augmented measurement consider Kalman filter (AMCKF) derivation.

Using the *a priori* definitions described in Eqs. (3.5) and (3.6), an augmented measurement equation can be defined as

$$\tilde{\boldsymbol{y}}_a = \begin{bmatrix} \tilde{\boldsymbol{y}} \\ \hat{\boldsymbol{x}}^- \\ \hat{\boldsymbol{p}}^- \end{bmatrix}, \qquad H_a = \begin{bmatrix} H_x & H_p \\ I & 0 \\ 0 & I \end{bmatrix}, \qquad \boldsymbol{v}_a = \begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{\eta} \\ \boldsymbol{\beta} \end{bmatrix} \tag{4.9}$$

$$\tilde{\boldsymbol{y}}_a = H_a \boldsymbol{z} + \boldsymbol{v}_a, \qquad \boldsymbol{v}_a \sim \mathrm{N}(0, R_a) \tag{4.10}$$

where

$$R_a = \begin{bmatrix} R & 0 & 0 \\ 0 & P_{xx}^- & P_{xp}^- \\ 0 & P_{px}^- & P_{pp}^- \end{bmatrix} = \begin{bmatrix} R & 0 \\ 0 & P_z^- \end{bmatrix} \tag{4.11}$$

Based on these definitions a least squares cost function, J_a, can be defined as

$$J_a = \frac{1}{2} \boldsymbol{\epsilon}_a^{\mathrm{T}} R_a^{-1} \boldsymbol{\epsilon}_a \tag{4.12}$$

where

$$\boldsymbol{\epsilon}_a = \tilde{\boldsymbol{y}}_a - H_a \hat{\boldsymbol{z}}^+ \tag{4.13}$$

The weighting matrix, R_a^{-1}, ensures the minimum variance measure of optimality is also achieved as described in Chapter II for least squares problems.

Solving for $\hat{\boldsymbol{z}}^+$ results in the classic least squares solution:

$$\hat{\boldsymbol{z}}^+ = \left(H_a^{\mathrm{T}} R_a^{-1} H_a \right)^{-1} H_a^{\mathrm{T}} R_a^{-1} \tilde{\boldsymbol{y}}_a \tag{4.14}$$

The covariance for the augmented state vector is

$$P_z^+ = E\left\{ \left(\hat{\boldsymbol{z}}^+ - \boldsymbol{z} \right) \left(\hat{\boldsymbol{z}}^+ - \boldsymbol{z} \right)^{\mathrm{T}} \right\} = \left(H_a^{\mathrm{T}} R_a^{-1} H_a \right)^{-1} \tag{4.15}$$

where

$$R_a^{-1} = \begin{bmatrix} R^{-1} & 0 & 0 \\ 0 & M_{xx}^- & M_{xp}^- \\ 0 & M_{px}^- & M_{pp}^- \end{bmatrix} \tag{4.16}$$

Using the fact that $R_a R_a^{-1} = I$, it can be shown that

$$M_{xx}^- = \left(P_{xx}^- - P_{xp}^- P_{pp}^{-^{-1}} P_{px}^-\right)^{-1} \tag{4.17}$$

$$M_{pp}^- = \left(P_{pp}^- - P_{px}^- P_{xx}^{-^{-1}} P_{xp}^-\right)^{-1} \tag{4.18}$$

$$M_{xp}^- = -M_{xx}^- P_{xp}^- P_{pp}^{-^{-1}} \tag{4.19}$$

$$M_{px}^- = M_{xp}^{-^{\mathrm{T}}} \tag{4.20}$$

which is the same as the results found in Chapter II.

Rewriting Eq. (4.14) as

$$\left(H_a^{\mathrm{T}} R_a^{-1} H_a\right) \hat{\boldsymbol{z}}^+ = H_a^{\mathrm{T}} R_a^{-1} \tilde{\boldsymbol{y}}_a \tag{4.21}$$

and multiplying out the matrix components results in

$$\begin{bmatrix} \left(H_x^{\mathrm{T}} R^{-1} H_x + M_{xx}^-\right) & \left(H_x^{\mathrm{T}} R^{-1} H_p + M_{xp}^-\right) \\ \left(H_p^{\mathrm{T}} R^{-1} H_x + M_{px}^-\right) & \left(H_p^{\mathrm{T}} R^{-1} H_p + M_{pp}^-\right) \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}^+ \\ \hat{\boldsymbol{p}}^+ \end{bmatrix} = \begin{bmatrix} H_x^{\mathrm{T}} R^{-1} \tilde{\boldsymbol{y}} + M_{xx}^- \hat{\boldsymbol{x}}^- + M_{xp}^- \hat{\boldsymbol{p}}^- \\ H_p^{\mathrm{T}} R^{-1} \tilde{\boldsymbol{y}} + M_{px}^- \hat{\boldsymbol{x}}^- + M_{pp}^- \hat{\boldsymbol{p}}^- \end{bmatrix} \tag{4.22}$$

After defining a few new variables, Eq. (4.22) reduces to

$$\begin{bmatrix} M_{xx}^+ & M_{xp}^+ \\ M_{px}^+ & M_{pp}^+ \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}^+ \\ \hat{\boldsymbol{p}}^+ \end{bmatrix} = \begin{bmatrix} N_x \\ N_p \end{bmatrix} \tag{4.23}$$

resulting in two equations for two unknowns, where

$$M_{xx}^+ = H_x^{\mathrm{T}} R^{-1} H_x + M_{xx}^- \tag{4.24}$$

$$M_{xp}^+ = H_x^{\mathrm{T}} R^{-1} H_p + M_{xp}^- \tag{4.25}$$

$$M_{px}^+ = H_p^{\mathrm{T}} R^{-1} H_x + M_{px}^- \tag{4.26}$$

$$M_{pp}^+ = H_p^{\mathrm{T}} R^{-1} H_p + M_{pp}^- \tag{4.27}$$

$$N_x = H_x^{\mathrm{T}} R^{-1} \tilde{\boldsymbol{y}} + M_{xx}^- \hat{\boldsymbol{x}}^- + M_{xp}^- \hat{\boldsymbol{p}}^- \tag{4.28}$$

$$N_p = H_p^{\mathrm{T}} R^{-1} \tilde{\boldsymbol{y}} + M_{px}^{-} \hat{\boldsymbol{x}}^{-} + M_{pp}^{-} \hat{\boldsymbol{p}}^{-} \tag{4.29}$$

Solving the first equation of Eq. 4.23 for $\hat{\boldsymbol{x}}^{+}$

$$\hat{\boldsymbol{x}}^{+} = M_{xx}^{+^{-1}} N_x - M_{xx}^{+^{-1}} M_{xp}^{+} \hat{\boldsymbol{p}}^{+} \tag{4.30}$$

By plugging this result into the second equation, $\hat{\boldsymbol{p}}^{+}$ is found to be

$$\hat{\boldsymbol{p}}^{+} = \left(M_{pp}^{+} - M_{px}^{+} M_{xx}^{+^{-1}} M_{xp}^{+} \right)^{-1} \left(N_p - M_{px}^{+} M_{xx}^{+^{-1}} N_x \right) \tag{4.31}$$

This result can be plugged directly into Eq. (4.30) to solve for $\hat{\boldsymbol{x}}^{+}$. The alternative is to solve the second equation of Eq. 4.23 for $\hat{\boldsymbol{p}}^{+}$ and plug it back into the first equation finding

$$\hat{\boldsymbol{x}}^{+} = \left(M_{xx}^{+} - M_{xp}^{+} M_{pp}^{+^{-1}} M_{px}^{+} \right)^{-1} \left(N_x - M_{xp}^{+} M_{pp}^{+^{-1}} N_p \right) \tag{4.32}$$

Using the fact that

$$M_z^{+} P_z^{+} = \begin{bmatrix} M_{xx}^{+} & M_{xp}^{+} \\ M_{px}^{+} & M_{pp}^{+} \end{bmatrix} \begin{bmatrix} P_{xx}^{+} & P_{xp}^{+} \\ P_{px}^{+} & P_{pp}^{+} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \tag{4.33}$$

the components of the updated covariance matrix are found to be

$$P_{xx}^{+} = \left(M_{xx}^{+} - M_{xp}^{+} M_{pp}^{+^{-1}} M_{px}^{+} \right)^{-1} \tag{4.34}$$

$$P_{pp}^{+} = \left(M_{pp}^{+} - M_{px}^{+} M_{xx}^{+^{-1}} M_{xp}^{+} \right)^{-1} \tag{4.35}$$

$$P_{xp}^{+} = -M_{xx}^{+^{-1}} M_{xp}^{+} P_{pp}^{+} \tag{4.36}$$

$$P_{px}^{+} = -M_{pp}^{+^{-1}} M_{px}^{+} P_{xx}^{+} = P_{xp}^{+^{\mathrm{T}}} \tag{4.37}$$

Using the Sherman-Morrison-Woodbury matrix inversion lemma (Appendix B) and setting

$$A = M_{xx}^{+}, \qquad B = -M_{xp}^{+}, \qquad C = M_{pp}^{+}, \qquad D = M_{px}^{+} \tag{4.38}$$

results in a new equation for P_{xx}^{+}

$$P_{xx}^{+} = M_{xx}^{+^{-1}} + M_{xx}^{+^{-1}} M_{xp}^{+} \left(M_{pp}^{+} - M_{px}^{+} M_{xx}^{+^{-1}} M_{xp}^{+} \right)^{-1} M_{px}^{+} M_{xx}^{+^{-1}} \tag{4.39}$$

Setting

$$P_{x}^{+} = M_{xx}^{+^{-1}} \tag{4.40}$$

$$S_{xp}^{+} = -M_{xx}^{+^{-1}} M_{xp}^{+} \tag{4.41}$$

allows Eq. (4.39) to be rewritten as

$$P_{xx}^{+} = P_{x}^{+} + S_{xp}^{+} P_{pp}^{+} S_{xp}^{+^{\mathrm{T}}} \tag{4.42}$$

Using Eq. (4.36), Eq. (4.41) can also be rewritten as

$$S_{xp}^{+} = P_{xp}^{+} P_{pp}^{+^{-1}} \tag{4.43}$$

The derivation of the AMCKF update equations begins by substituting Eq. (4.28) into Eq. (4.30) to produce

$$\hat{\boldsymbol{x}}^{+} = M_{xx}^{+^{-1}} \left(H_{x}^{\mathrm{T}} R^{-1} \tilde{\boldsymbol{y}} + M_{xx}^{-} \hat{\boldsymbol{x}}^{-} + M_{xp}^{-} \hat{\boldsymbol{p}}^{-} \right) - M_{xx}^{+^{-1}} M_{xp}^{+} \hat{\boldsymbol{p}}^{+} \tag{4.44}$$

As with the MVCKF, parameter estimates and their associated covariance remain unchanged by implementing the constant parameter constraint (CPC). Tapley *et al.* go a step further and assume the *a priori* cross-covariance, P_{xp}^{-}, is also zero [8]. While this is a plausible assumption for a batch estimator, it certainly does not hold for more general cases especially sequential estimation, which is the goal here. By imposing the CPC, Eq. (4.44) is rewritten as

$$\hat{\boldsymbol{x}}^{+} = M_{xx}^{+^{-1}} \left(H_{x}^{\mathrm{T}} R^{-1} \tilde{\boldsymbol{y}} + M_{xx}^{-} \hat{\boldsymbol{x}}^{-} + M_{xp}^{-} \hat{\boldsymbol{p}} \right) - M_{xx}^{+^{-1}} M_{xp}^{+} \hat{\boldsymbol{p}} \tag{4.45}$$

Substituting in the values of both and M_{xx}^{-} and M_{xp}^{-} from Eqs. (4.17) and (4.19), respectively, yields

$$\hat{\boldsymbol{x}}^{+} = \hat{\boldsymbol{x}}^{-} + K \left(\tilde{\boldsymbol{y}} - H_{x} \hat{\boldsymbol{x}}^{-} - H_{p} \hat{\boldsymbol{p}} \right) \tag{4.46}$$

after simplifying. The gain, K, is defined as

$$K = M_{xx}^{+^{-1}} H_x^{\mathrm{T}} R^{-1} = P_x^{+} H_x^{\mathrm{T}} R^{-1} \tag{4.47}$$

The second equation results from substituting in the definition of P_x^+ from Eq. (4.40).

By taking the inverse of Eq. (4.24) and using the Sherman-Morrison-Woodbury matrix inversion lemma, once again, where

$$A = M_{xx}^{-}, \qquad B = H_x^{\mathrm{T}}, \qquad C = R^{-1}, \qquad D = H_x \tag{4.48}$$

the partial state covariance, P_x^+ can be rewritten as

$$P_x^{+} = M_{xx}^{-^{-1}} - M_{xx}^{-^{-1}} H_x^{\mathrm{T}} \left(H_x M_{xx}^{-^{-1}} H_x^{\mathrm{T}} + R \right)^{-1} H_x M_{xx}^{-^{-1}} \tag{4.49}$$

Plugging this into Eq. (4.47) gives

$$K = M_{xx}^{-^{-1}} H_x^{\mathrm{T}} \left[I - \left(H_x M_{xx}^{-^{-1}} H_x^{\mathrm{T}} + R \right)^{-1} H_x M_{xx}^{-^{-1}} H_x^{\mathrm{T}} \right] R^{-1} \tag{4.50}$$

Pulling out $\left(H_x M_{xx}^{-^{-1}} H_x^{\mathrm{T}} + R \right)^{-1}$ from the left further reduces the equation to

$$K = M_{xx}^{-^{-1}} H_x^{\mathrm{T}} \left(H_x M_{xx}^{-^{-1}} H_x^{\mathrm{T}} + R \right)^{-1} \tag{4.51}$$

where $M_{xx}^{-^{-1}}$ is found by taking the inverse of Eq. (4.17). This value of the gain is put back into Eq. (4.49) resulting in

$$P_x^{+} = M_{xx}^{-^{-1}} - K H_x M_{xx}^{-^{-1}} \tag{4.52}$$

Using the inverse of Eq. (4.17) and simplifying, results in an equation for P_x^+ based on the propagated covariance components

$$P_x^{+} = (I - K H_x) \left(P_{xx}^{-} - P_{xp}^{-} P_{pp}^{-1} P_{px}^{-} \right) \tag{4.53}$$

Expanding out the updated cross-covariance, P_{xp}^+, described in Eq. (4.36) using the definition of M_{xp}^+ produces

$$P_{xp}^{+} = -P_x^{+} \left(H_x^{\mathrm{T}} R^{-1} H_p + M_{xp}^{-} \right) P_{pp} \tag{4.54}$$

If the definition of M_{xx}^{+} is plugged into Eq. (4.19) and replaces M_{xp}^{-} in the above equation, the new result is

$$P_{xp}^{+} = P_{x}^{+}\left[\left(M_{xx}^{+} - H_{x}^{\mathrm{T}}R^{-1}H_{x}\right)P_{xp}^{-}P_{pp}^{-1} - H_{x}^{\mathrm{T}}R^{-1}H_{p}\right]P_{pp} \tag{4.55}$$

Simplifying further

$$P_{xp}^{+} = \left(I - KH_{x}\right)P_{xp}^{-} - KH_{p}P_{pp} \tag{4.56}$$

Note that this is the same result found by the MVCKF in Eq. (3.28). Based on the definition of the sensitivity matrix, S_{xp}^{+}, from Eq. (4.43) it is also apparent that

$$S_{xp}^{+} = \left(I - KH_{x}\right)P_{xp}^{-}P_{pp}^{-1} - KH_{p} = P_{xp}^{+}P_{pp}^{-1} \tag{4.57}$$

which verifies the result in Eq. (4.43). The updated state covariance is known from Eq. (4.42) in the consider analysis derivation to be

$$P_{xx}^{+} = P_{x}^{+} + S_{xp}^{+}P_{pp}S_{xp}^{+^{\mathrm{T}}} \tag{4.58}$$

Note that this derivation is different than the sequential estimator given by Tapley *et al.* [8]. The above derivation propagates and updates the considered state values and covariance directly. The set of equations presented by Tapley *et al.*, however, uses the traditional Kalman filter to propagate and update the state values and then applies an additional update to calculate the considered state values and covariance [8]. This approach is consistent with what is known as consider analysis [17]. Consider filtering on the other hand, directly uses the consider states in the filter itself as is done in both the MVCKF and the AMCKF [17]. The AMKCF is summarized in Table 15.

a. CKF Comparison

At first glance, the two CKF developments appear similar save for the state covariance update equations, P_{xx}^{+}, and the gains, K. Based on these differences, the

Table 15: Discrete Augmented Measurement Consider Kalman Filter

Model	$\boldsymbol{x}_{k+1} = \Phi_k \boldsymbol{x}_k + \Psi_k \boldsymbol{p} + \Gamma_k \boldsymbol{w}_k, \qquad \boldsymbol{w}_k \sim \mathrm{N}(0, Q_k)$ $\tilde{\boldsymbol{y}}_k = H_{x_k} \boldsymbol{x}_k + H_{p_k} \boldsymbol{p} + \boldsymbol{v}_k, \qquad \boldsymbol{v}_k \sim \mathrm{N}(0, R_k)$
Gain	$M_{xx}^{-} = \left(P_{xx}^{-} - P_{xp}^{-} P_{pp}^{-1} P_{px}^{-}\right)^{-1}$ $K = M_{xx}^{-^{-1}} H_x^{\mathrm{T}} \left(H_x M_{xx}^{-^{-1}} H_x^{\mathrm{T}} + R\right)^{-1}$
Update	$\hat{\boldsymbol{x}}^{+} = \hat{\boldsymbol{x}}^{-} + K\left(\tilde{\boldsymbol{y}} - H_x \hat{\boldsymbol{x}}^{-} - H_p \hat{\boldsymbol{p}}\right)$ $P_x^{+} = \left(I - K H_x\right)\left(P_{xx}^{-} - P_{xp}^{-} P_{pp}^{-1} P_{px}^{-}\right)$ $S_{xp}^{+} = \left(I - K H_x\right) P_{xp}^{-} P_{pp}^{-1} - K H_p$ $P_{xx}^{+} = P_x^{+} + S_{xp}^{+} P_{pp} S_{xp}^{+^{\mathrm{T}}}$ $P_{xp}^{+} = \left(I - K H_x\right) P_{xp}^{-} - K H_p P_{pp}$
Propagation	$\hat{\boldsymbol{x}}_{k+1}^{-} = \Phi_k \hat{\boldsymbol{x}}_k^{+} + \Psi_k \hat{\boldsymbol{p}}$ $P_{xx,k+1}^{-} = \Phi_k P_{xx,k}^{+} \Phi_k^{\mathrm{T}} + \Phi_k P_{xp,k}^{+} \Psi_k^{\mathrm{T}} + \Psi_k P_{px,k}^{+} \Phi_k^{\mathrm{T}}$ $+ \Psi_k P_{pp} \Psi_k^{\mathrm{T}} + \Gamma_k Q_k \Gamma_k^{\mathrm{T}}$ $P_{xp,k+1}^{-} = \Phi_k P_{xp,k}^{+} + \Psi_k P_{pp}$

question is whether the two methods are in fact equivalent and, if not, are there occasions when they are? A better understanding of the difference between the two CKFs is desired through a comparison of their defining equations.

To compare their state covariance update equations, Eqs. (4.53) and (4.57) are plugged into Eq. (4.42). Following a reconsolidation of terms it is found that

$$P_{xx}^{+} = P_{xx}^{-} - K\left(H_x P_{xx}^{-} + H_p P_{px}^{-}\right) - \left(P_{xp}^{-} P_{pp}^{-1} P_{px}^{-} H_x^{\mathrm{T}} + P_{xp}^{-} H_p^{\mathrm{T}}\right) K^{\mathrm{T}} + K\left(H_x P_{xp}^{-} P_{pp}^{-1} P_{px}^{-} H_x^{\mathrm{T}} + + H_x P_{xp}^{-} H_p^{\mathrm{T}} + H_p P_{px}^{-} H_x^{\mathrm{T}} + H_p P_{pp} H_p^{\mathrm{T}}\right) K^{\mathrm{T}} \quad (4.59)$$

This equation is strikingly similar to Eq. (3.27) except for the $P_{xp}^{-} P_{pp}^{-1} P_{px}^{-}$ terms and a lack of the measurement covariance, R. Plugging Eq. (4.17) into the AMCKF gain function, Eq. (4.51), yields

$$K\left(H_x P_{xx}^{-} H_x^{\mathrm{T}} - H_x P_{xp}^{-} P_{pp}^{-1} P_{px}^{-} H_x^{\mathrm{T}} + R\right) = P_{xx}^{-} H_x^{\mathrm{T}} - P_{xp}^{-} P_{pp}^{-1} P_{px}^{-} H_x^{\mathrm{T}} \quad (4.60)$$

Pulling both $P_{xp}^{-} P_{pp}^{-1} P_{px}^{-}$ terms onto one side and postmultiplying by K^{T} gives

$$K H_x P_{xp}^{-} P_{pp}^{-1} P_{px}^{-} H_x^{\mathrm{T}} K^{\mathrm{T}} - H_x P_{xp}^{-} P_{pp}^{-1} P_{px}^{-} H_x^{\mathrm{T}} K^{\mathrm{T}} = K\left(H_x P_{xx}^{-} H_x^{\mathrm{T}} + R\right) K^{\mathrm{T}} - P_{xx}^{-} H_x^{\mathrm{T}} K^{\mathrm{T}} \quad (4.61)$$

Plugging this result back into Eq. (4.59) shows that the two state covariance update equations are in fact equivalent only with different gain values. Thus, not surprisingly, it is the calculation of the gain that controls the equivalence of both derivations.

To compare the gain equations, rearrange Eq. (4.60) to produce

$$K = \left(P_{xx}^{-} H_x^{\mathrm{T}} - P_{xp}^{-} P_{pp}^{-1} P_{px}^{-} H_x^{\mathrm{T}}\right)\left(H_x P_{xx}^{-} H_x^{\mathrm{T}} - H_x P_{xp}^{-} P_{pp}^{-1} P_{px}^{-} H_x^{\mathrm{T}} + R\right)^{-1} \quad (4.62)$$

By looking at the first set of parentheses for both Eq. (3.31) and (4.62), it is clear that for the two gains to be comparable

$$H_p^{\mathrm{T}} = -P_{pp}^{-1} P_{px}^{-} H_x^{\mathrm{T}} \quad (4.63)$$

Premultiplying Eq. (4.63) by P_{pp} and postmultiplying by $R^{-1}H_x$ generates

$$P_{pp}H_p^{\mathrm{T}}R^{-1}H_x = -P_{px}^{-}H_x^{\mathrm{T}}R^{-1}H_x \tag{4.64}$$

or

$$P_{px}^{-} = -P_{pp}H_p^{\mathrm{T}}R^{-1}H_x\left(H_x^{\mathrm{T}}R^{-1}H_x\right)^{-1} \tag{4.65}$$

Thus, in the case when Eq. (4.65) is true, then Eq. (4.63) also holds. Pre-multiplying Eq. (4.63) by P_{xp}^{-} gives

$$P_{xp}^{-}H_p^{\mathrm{T}} = -P_{xp}^{-}P_{pp}^{-1}P_{px}^{-}H_x^{\mathrm{T}} \tag{4.66}$$

The second set of parentheses in Eq. (4.62) can be rewritten as

$$\begin{aligned} &H_xP_{xx}^{-}H_x^{\mathrm{T}} - H_xP_{xp}^{-}P_{pp}^{-1}P_{xp}^{-}H_x^{\mathrm{T}} + R \\ &= H_xP_{xx}^{-}H_x^{\mathrm{T}} - H_xP_{xp}^{-}P_{pp}^{-1}P_{px}^{-}H_x^{\mathrm{T}} - H_xP_{xp}^{-}P_{pp}^{-1}P_{px}^{-}H_x^{\mathrm{T}} - H_xP_{xp}^{-}P_{pp}^{-1}P_{pp}P_{pp}^{-1}P_{px}^{-}H_x^{\mathrm{T}} + R \end{aligned} \tag{4.67}$$

Using Eq. (4.63) four times and the fact that $P_{xp}^{-} = P_{px}^{-^{\mathrm{T}}}$ in the above reduces the righthand side to

$$\begin{aligned} &H_xP_{xx}^{-}H_x^{\mathrm{T}} - H_xP_{xp}^{-}P_{pp}^{-1}P_{xp}^{-}H_x^{\mathrm{T}} + R \\ &\qquad = H_xP_{xx}^{-}H_x^{\mathrm{T}} + H_xP_{xp}^{-}H_p^{\mathrm{T}} + H_pP_{px}^{-}H_x^{\mathrm{T}} + H_pP_{pp}H_p^{\mathrm{T}} + R \end{aligned} \tag{4.68}$$

which is equivalent to the inverted portion of the MVCKF gain. From this analysis it is clear that when either Eq. (4.63) or (4.65) holds for all time, the two CKFs are equivalent, otherwise the AMCKF is not a minimum variance filter.

Taking another look at the AMCKF begs the question as to whether the augmented measurement structure can also produce the optimal gain. Instead of imposing the CPC at Eq. (4.30), begin with Eq. (4.32). In this equation the updated state, $\hat{\boldsymbol{x}}^{+}$, is given as a function of the measurements, $\tilde{\boldsymbol{y}}$, the *a priori* state estimates,

$\hat{\boldsymbol{x}}^-$, and the *a priori* parameter estimates, $\hat{\boldsymbol{p}}$, because of the definitions of N_x and N_p given by Eqs. (4.28) and (4.29), respectively. Plugging in these equations along with Eqs. (4.24) - (4.27) for M_{xx}^-, M_{xp}^-, M_{px}^-, and M_{pp}^-, respectively, results in

$$\begin{aligned}\hat{\boldsymbol{x}}^+ = &\left(M_{xx}^+ - M_{xp}^+ M_{pp}^{+^{-1}} M_{px}^+\right)^{-1} \Big[\left(M_{xx}^+ - M_{xp}^+ M_{pp}^{+^{-1}} M_{px}^+\right)\hat{\boldsymbol{x}}^- \\ &+ \left(H_x^{\mathrm{T}} - M_{xp}^+ M_{pp}^{+^{-1}} H_p^{\mathrm{T}}\right) R^{-1}\tilde{\boldsymbol{y}} - \left(H_x^{\mathrm{T}} - M_{xp}^+ M_{pp}^{+^{-1}} H_p^{\mathrm{T}}\right) R^{-1} H_x \hat{\boldsymbol{x}}^- \\ &- \left(H_x^{\mathrm{T}} - M_{xp}^+ M_{pp}^{+^{-1}} H_p^{\mathrm{T}}\right) R^{-1} H_p \hat{\boldsymbol{p}}\Big] \end{aligned} \tag{4.69}$$

By simplifying the above equation and using the definitions of P_{xx}^+ and P_{xp}^+ given by Eqs. (4.34) and (4.36), respectively, gives

$$\hat{\boldsymbol{x}}^+ = \hat{\boldsymbol{x}}^- + \left(P_{xx}^+ H_x^{\mathrm{T}} + P_{xp}^+ H_p^{\mathrm{T}}\right) R^{-1}\left(\tilde{\boldsymbol{y}} - H_x\hat{\boldsymbol{x}}^- - H_p\hat{\boldsymbol{p}}\right) \tag{4.70}$$

and we can define the gain as

$$K = \left(P_{xx}^+ H_x^{\mathrm{T}} + P_{xp}^+ H_p^{\mathrm{T}}\right) R^{-1} \tag{4.71}$$

This is the same gain as that found in Eq. (2.72) for sequential least squares. The only difference here is that the usual substitution of $W = R^{-1}$ has been made to produce the optimal covariance. Therefore, it is verified that the augmented measurement and minimum variance derivations are equivalent provided that the CPC is not prematurely enforced. Using the *a posteriori* covariances is a simpler comparison than using the *a priori* covariances. The latter requires significant use of the Sherman-Morrison-Woodbury matrix inversion lemma.

2. The Minimum Variance Reduced-order Filter

Another reduced-order method is called the minimum variance reduced-order filter (MVROF) [12]. The MVROF applies a transformation matrix, T, to the augmented state vector, $\boldsymbol{z}$, thereby directly reducing the order of the filter. This method

works well for parameters in the measurement equations, but information is lost for dynamic model parameters. This information loss produces a bias in the state estimates in the general case [38, 39]. The filter also requires a pseudo-inverse which can be computationally expensive if the transformation matrix varies with time [40].

The MVROF transformation is described mathematically by

$$\boldsymbol{x} = T\boldsymbol{z} \tag{4.72}$$

Typically, $\boldsymbol{x}$ will be a subset or a combination of a subset of the vector $\boldsymbol{z}$. For this reason, the pseudo-inverse of the transformation matrix must also be used where

$$TT^{\dagger} = I_{n\times n} \tag{4.73}$$

since the length of the reduced vector, $\boldsymbol{x}$, is n. It should be noted, however, that $T^{\dagger}T$ is a square matrix that is not the identity matrix. Hutchinson *et al.* provide a complete derivation of the MVROF [12]. A summary of the MVROF's components is shown in Table 16.

3. Example IV.2

The freefall scenario used in Example IV.1 is used again here to compare the AMCKF and the MVROF to the MVCKF. The freefall example is used because the solitary dynamic parameter appears linearly with the states. The TKF is also included as a base for comparison. The system structures for each filter are equivalent and can be found in Example IV.1.

A set of 10000 Monte Carlo simulations was run for all four filters. Process noise was not used in any of the filters. The true values of the initial parameters and gravity constant are shown in Table 17. The table also contains the standard deviations for each parameter used to determine initial estimates for each simulation. The gravity

Table 16: Discrete Minimum Variance Reduced-order Filter

Model	$\boldsymbol{z}_{k+1} = \Theta_k \boldsymbol{z}_k + \Upsilon_k \boldsymbol{w}_k, \qquad \boldsymbol{w}_k \sim \mathrm{N}(0, Q_k)$ $\tilde{\boldsymbol{y}}_k = H_{z_k} \boldsymbol{z}_k + \boldsymbol{v}_k, \qquad \boldsymbol{v}_k \sim \mathrm{N}(0, R_k)$ $\boldsymbol{x}_k = T \boldsymbol{z}_k, \qquad TT^\dagger = I$
Gain	$K = TP_z^- H_z^{\mathrm{T}} \left(H_z P_z^- H_z^{\mathrm{T}} + R\right)^{-1}$
Update	$\hat{\boldsymbol{x}}^+ = \hat{\boldsymbol{x}}^- + K\left(\tilde{\boldsymbol{y}} - H_z T^\dagger \hat{\boldsymbol{x}}^-\right)$ $P_z^+ = \left(I - T^\dagger K H_z\right) P_z^- \left(I - T^\dagger K H_z\right)^{\mathrm{T}} + T^\dagger K R K^{\mathrm{T}} \left(T^\dagger\right)^{\mathrm{T}}$
Propagation	$\hat{\boldsymbol{x}}_{k+1}^- = T\Theta_k T^\dagger \hat{\boldsymbol{x}}_k^+$ $P_{z,k+1}^- = \Theta_k P_{z,k}^+ \Theta_k^{\mathrm{T}} + \Upsilon_k Q_k \Upsilon_k^{\mathrm{T}}$

constant is small to simulate freefall on a large asteroid. The diagonals of the initial state and parameter covariances, P_{xx_0} and P_{pp}, were set to the square of the standard deviations shown in Table 17. This allowed the initial error to be well within the initial 3σ bound for each state. All initial covariances off the diagonal were set to zero. The measurement standard deviation was set to 2 km for both measurements and measurements were collected at a sampling rate of 1 Hz.

Table 17: State and Parameter Values Used in Example IV.2

	x_0 (km)	$\dot{x}_0$ (km/s)	y_0 (km)	$\dot{y}_0$ (km/s)	g (m/s^2)
Truth	0	0.3	50	0	0.220
Stan. Dev.	2	0.05	1	0.02	0.045

The initial cross-covariance between the states and parameters, P_{xp_0}, was set to zero for the MVROF and the MVCKF. Based upon the analysis performed in Section 1 it is clear that an initial estimate of the cross-covariance should be given. As a result, the steady-state value of the cross-covariance in the MVCKF was used to calculate the initial value for the AMCKF, which is

$$P_{xp_0} = \begin{bmatrix} 0 & -7e\text{-}6 & 0 & -2e\text{-}7 \end{bmatrix}^{\mathrm{T}} \tag{4.74}$$

To effectively compare the MVROF to the other filters, the transformation matrix, T, was chosen to be

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.75}$$

where $T^{\dagger} = T^{\mathrm{T}}$. Other values of T were also tested, but results showed that this selection produced the best results.

4. Single Run Results

Figures 28 - 31 contains the results of the final simulation from the Monte Carlo trials. The X position and velocity values are not shown because these values are unaffected by the uncertainty in the dynamic parameter. The TKF has no way to account for the incorrect parameter, thus the error in both position and velocity increase with time and the covariance is unable to capture it. The MVROF has a very distinct bias in these results for both position and velocity. The velocity error is completely outside of the 3σ bound in Figure 30(b). Comparing the MVROF covariance estimates to that of the MVCKF finds that they are equivalent. This is expected since our particular choice of T causes the covariance update and propaga-

tion equations to be the same for the two filters. Next, comparing the state estimates of both the MVROF and the MVCKF shows that they are similar, but differ by an unknown bias. Plotting the differences between their state estimates, as in Figure 32, shows that while there are some initial fluctuations, when the covariance reaches a steady-state so does the bias between the two filters.

The results of the AMCKF show that this filter clearly diverges even with nonzero values for P_{xp_0}. Moreover, despite derivations based on a similar principle, the two CKFs behave quite differently in practice. The AMCKF suffers the same state error problems that the TKF shows, but it does manage to capture the error within the covariance bounds. The problem is that the filter clearly diverges for both position and velocity in the Y direction. The similarity between the state estimates of the TKF and the AMCKF occurs for an entirely different reason than the similarity between the MVCKF and the AKF in Chapter II. For the state estimates to be equal, the gains for both filters must be equivalent ($M_{xx}^{-^{-1}} = P_{xx}^{-}$). While it is clear that this is true for the initial case where $P_{xp_0}^{-} = \mathbf{0}$, this must hold for all time. Examining how $M_{xx}^{-^{-1}}$ is propagated in the discrete case, we find that

$$P_{xp,k+1}^{-} P_{pp}^{-1} P_{px,k+1}^{-} = \Phi_k P_{xp,k}^{+} P_{pp}^{-1} P_{px,k}^{+} \Phi_k^{\mathrm{T}} + \Phi_k P_{xp,k}^{+} \Psi_k^{\mathrm{T}} + \Psi_k P_{px,k}^{+} \Phi_k^{\mathrm{T}} + \Psi_k P_{pp} \Psi_k^{\mathrm{T}} \quad (4.76)$$

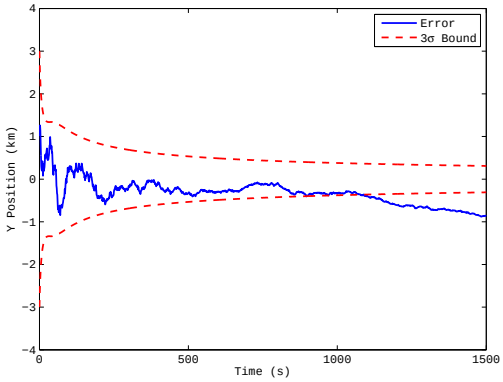
from Eq. (3.38). By assuming process noise is zero, and plugging this result and Eq. (3.37) back into the inverse of Eq. (4.17) gives

$$P_{xx,k}^{-} - P_{xp,k}^{-} P_{pp}^{-1} P_{px,k}^{-} = \Phi_k \left(P_{xx,k}^{+} - P_{xp,k}^{+} P_{pp}^{-1} P_{px,k}^{+} \right) \Phi_k^{\mathrm{T}} \quad (4.77)$$

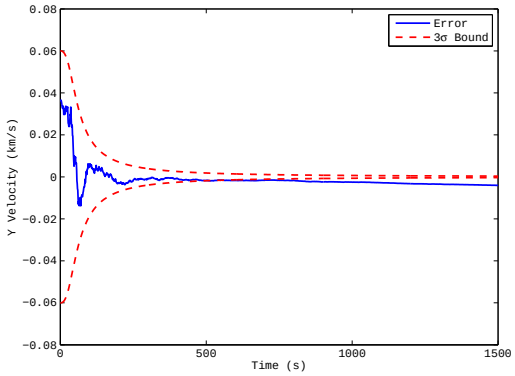
Thus, $M_{xx}^{-^{-1}}$ is simply the propagated form of the covariances' updated values. This can also be shown for the continuous case.

Keeping Eq.(4.77) in mind, the discrete update equation is now explored. Plugging in Eq. (4.53) into Eq. (4.42) yields

$$P_{xx}^{+} = (I - K H_x) \left(P_{xx}^{-} - P_{xp}^{-} P_{pp}^{-1} P_{px}^{-} \right) + S_{xp}^{+} P_{pp} S_{xp}^{+^{\mathrm{T}}} \quad (4.78)$$

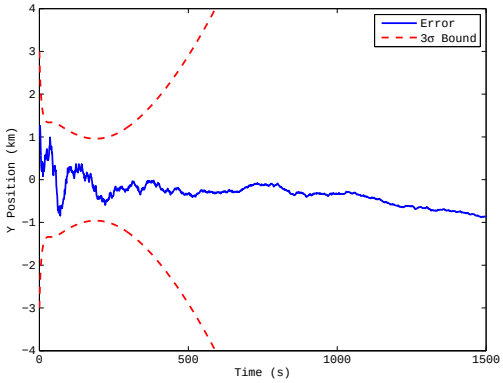


(a) Position

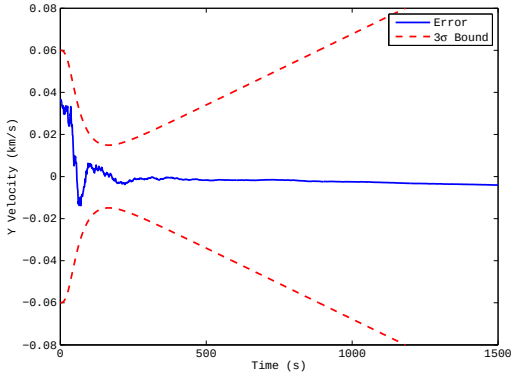


(b) Velocity

Fig. 28: Y Position and Velocity Errors with 3σ Bounds for the TKF

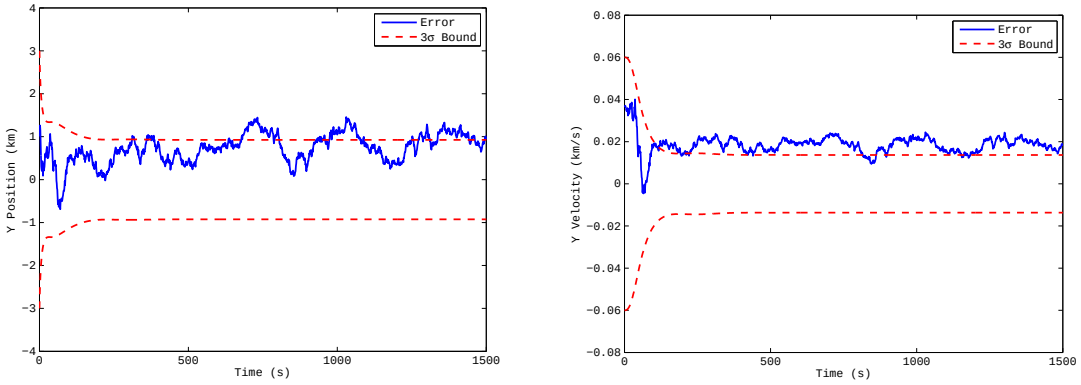


(a) Position



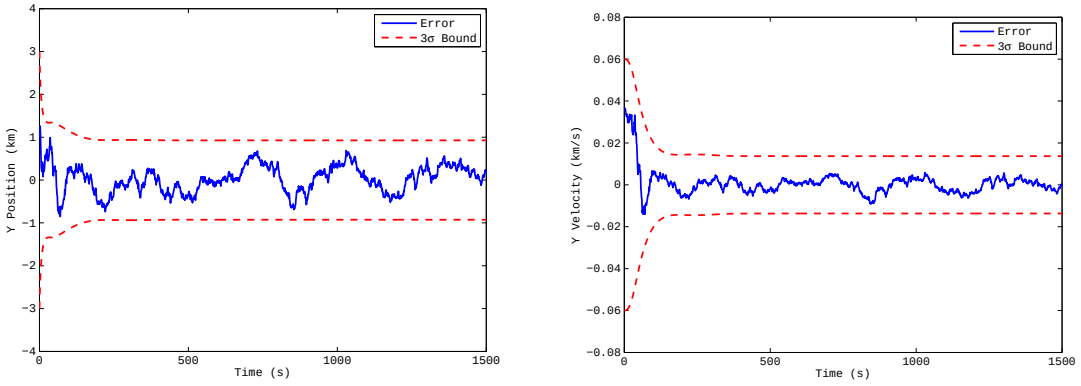
(b) Velocity

Fig. 29: Y Position and Velocity Errors with 3σ Bounds for the AMCKF



(a) Position (b) Velocity

Fig. 30: Y Position and Velocity Errors with 3σ Bounds for the MVROF



(a) Position (b) Velocity

Fig. 31: Y Position and Velocity Errors with 3σ Bounds for the MVCKF

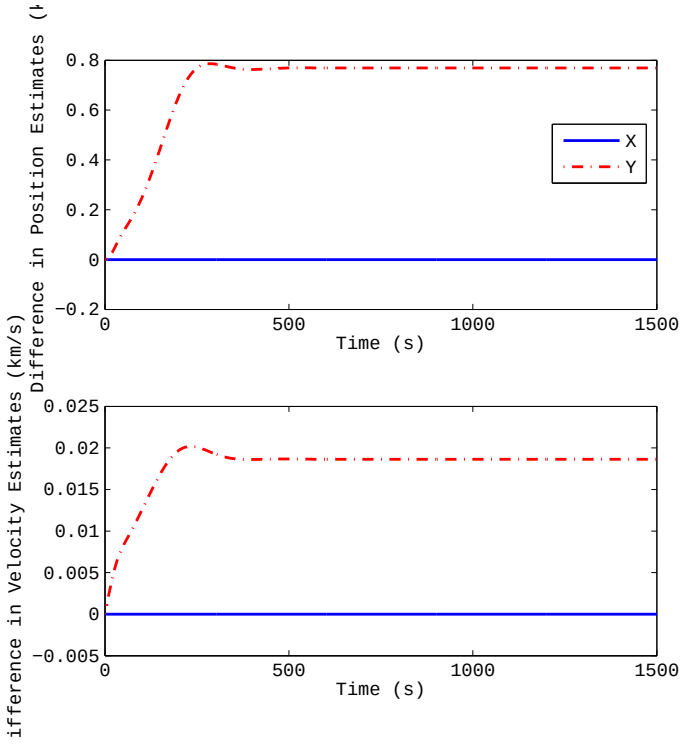


Fig. 32: Difference Between MVROF and MVCKF State Estimates

From Eq. (4.57) it can be shown that $S^{+}_{xp}P_{pp}S^{+^{\mathrm{T}}}_{xp} = P^{+}_{xp}P^{-1}_{pp}P^{+}_{px}$. Subtracting this result from both sides of Eq. (4.78) produces

$$P^{+}_{xx} - P^{+}_{xp}P^{-1}_{pp}P^{+}_{px} = (I - KH_x)\left(P^{-}_{xx} - P^{-}_{xp}P^{-1}_{pp}P^{-}_{px}\right) \tag{4.79}$$

While this may not seem like much at first, combining this result with Eq. (4.77) gives

$$M^{-^{-1}}_{xx,k+1} = \Phi_k\left((I - KH_x)\,M^{-^{-1}}_{xx,k}\right)\Phi^{\mathrm{T}}_k \tag{4.80}$$

In the special case where $P^{-}_{xp_0} = 0$, the initial value of $M^{-^{-1}}_{xx}$ is $P^{-}_{xx_0}$. Thus, in this special case the gain of the AMCKF is equivalent to that of the TKF and so are their state estimates. The difference between the state estimates of the TKF and the AMCKF are shown in Figure 33. Differences do occur between the two filters early on because of the non-zero initial cross-covariance estimates, notice that the AMCKF

state estimates eventually converge towards the TKF estimates. Moreover, this quick exercise shows that the initial value of P_{xp} must be reasonably well known or it must be tuned for the AMCKF to work effectively.

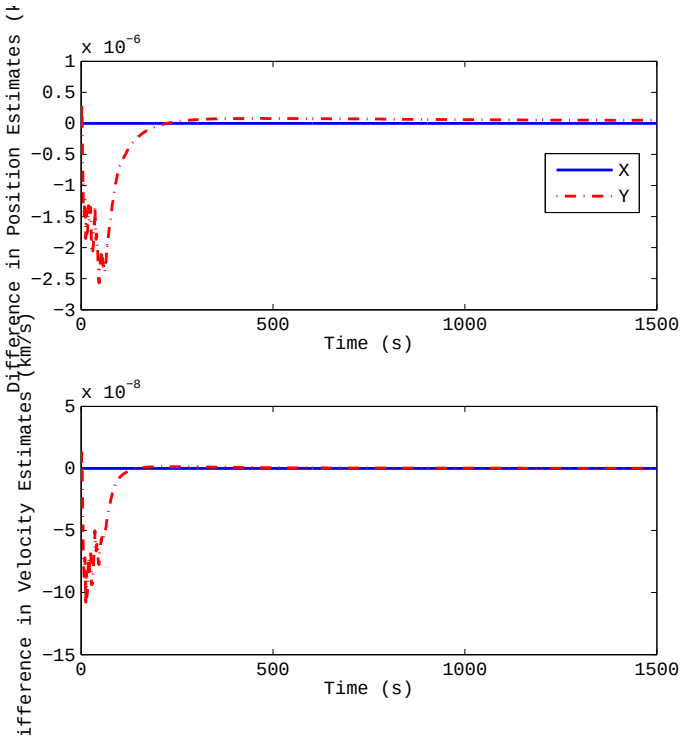


Fig. 33: Difference Between TKF and AMCKF State Estimates

5. Monte Carlo Results

The results of the Monte Carlo (MC) analysis for the error in the final y position and velocity estimates of all four filters are shown in Tables 18 and 19, respectively. The results of the fully augmented Kalman filter (AKF) have also been included as reference values. The statistical mean and standard deviation of the Monte Carlo results are given as well as the standard deviation as predicted by linear error theory (LET).

Table 18: $y(t_f)$ Position Error Statistics from both Monte Carlo (MC) and Linear Error Theory (LET) for Steady-State P_{xp_0} Initial Values

	MC Mean (km)	MC Stan. Dev. (km)	LET Stan. Dev. (km)
TKF	0.0187	8.439	0.1031
AMCKF	0.0187	8.439	8.467
MVROF	0.7473	0.2668	0.3085
MVCKF	-0.0020	0.3069	0.3085
AKF	-0.0027	0.1550	0.1545

Table 19: $\dot{y}(t_f)$ Velocity Error Statistics from both Monte Carlo (MC) and Linear Error Theory (LET) for Steady-State P_{xp_0} Initial Values

	MC Mean (m/s)	MC Stan. Dev. (m/s)	LET Stan. Dev. (m/s)
TKF	0.0771	33.70	0.1188
AMCKF	0.0771	33.71	33.82
MVROF	18.16	2.630	4.559
MVCKF	0.0039	4.523	4.559
AKF	-0.0085	0.4769	0.4747

These results verify that the TKF, AKF, AMCKF, and MVCKF are all unbiased estimators. The AMCKF means are slightly larger than the other three, but the observed increase is due to the large variance of the filter. The bias observed in the MVROF results confirms that the biases observed in the Single Run results (Figures 30(a) and 30(b)) were not a fluke, but consistent results for all trials. This result also validates the theoretical bias in the MVROF as predicted by Asher *et al.* [38]. While the statistical standard deviation is close to the predicted standard

deviation for the AMCKF, both values are quite large resulting from the divergent nature of the filter. Both the AMCKF and the MVCKF accurately calculate the standard deviation from linear error theory, but the MVCKF does not exhibit the divergent behavior found in the AMCKF. The AKF outperforms all the other filters as expected since the dynamic parameter is completely observable. It should also be noted that the TKF statistical standard deviation is drastically less than the predicted value, implying that the estimated covariance is more confident than it should be in its results by a significant degree.

To analyze the effect of the initial value of P_{xp_0} on the process noise, an additional set of simulations was performed assuming the initial cross-covariance was

$$P_{xp_0} = \begin{bmatrix} 0 & -7e\text{-}8 & 0 & -2e\text{-}9 \end{bmatrix}^{\mathrm{T}} \tag{4.81}$$

The results of these runs are shown in Tables 20 and 21 and all four filters exhibit similar results to those observed in Tables 18 and 19.

Table 20: $y(t_f)$ Position Error Statistics from both Monte Carlo (MC) and Linear Error Theory (LET) for Reduced P_{xp_0} Initial Values

	MC Mean (km)	MC Stan. Dev. (km)	LET Stan. Dev. (km)
TKF	-0.0058	8.445	0.1031
AMCKF	-0.0058	8.445	8.467
MVROF	0.7455	0.2674	0.3085
MVCKF	-0.0041	0.3083	0.3085
AKF	-0.0007	0.1530	0.1545

Table 21: $\dot{y}(t_f)$ Velocity Error Statistics from both Monte Carlo (MC) and Linear Error Theory (LET) for Reduced P_{xp_0} Initial Values

	MC Mean (m/s)	MC Stan. Dev. (m/s)	LET Stan. Dev. (m/s)
TKF	-0.0190	33.73	0.1188
AMCKF	-0.0190	33.73	33.82
MVROF	18.13	2.643	4.559
MVCKF	-0.0297	4.555	4.559
AKF	0.0015	0.4705	0.4747

CHAPTER V

HARDWARE APPLICATIONS

Up until now, focus has been placed on theoretical developments and their verification through the use of simple examples. To truly test the validity of the concepts developed in the earlier chapters, the consider methodology must be applied to hardware applications and compensate for real world deviations. This chapter explores two hardware applications and how the consider methodology behaves with both of them. The first example analyzes the use of a filter on the odometry of a three degree-of-freedom (DOF) omni-directional robot base developed at Texas A&M University. The second example explores the application of the consider framework to an attitude filter of a stellar positioning system (SPS) also developed at Texas A&M.

A. Single Wheel Odometry

The Land, Air, and Space Robotics (LASR) Laboratory at Texas A&M University has recently developed a novel robotic platform designed to improve simulation hardware for satellite proximity operations. The holonomic omni-directional motion emulation robot (HOMER) consists of a mobile, planar base accompanied by a state of the art Alio Stewart platform to provide full 6 DOF motion for multi-vehicle operations. The robotic base (Figure 34) uses a three castor design to provide omni-directional planar motion on any flat surface. A configuration known as an Active Split-Offset Castor (ASOC) was used to provide each individual castor with the capability to move smoothly in any planar direction by independently driving each of its two wheels. The general ASOC concept is shown in Figure 35. Three of these

castors are necessary for stability and all six wheels are controlled to provide even distribution of motor torques for all possible trajectories [41, 42].

Each wheel on the HOMER Base is also equipped with an incremental optical encoder. These encoders are used to record wheel rotations, which can then be used to integrate the position of the wheels to provide an inertial position estimate. This process is known as odometry. Although odometry can provide an accurate measure of relative distance traveled, it is still subject to a number of systematic and non-systematic errors that will cause position estimate errors. Systematic errors result from uncertainties in the wheel base and from wheels of different sizes. Non-systematic errors result from how the wheels interact with the floor including but not limited to wheel slippage and variations in the flatness of the floor [43]. These errors cause the position estimate to drift not with time, but as a function of distance traveled [41]. Using a method he developed called internal position error correction (IPEC), Borenstein was able to reduce the error in his odometry measurements by an order of magnitude over conventional wheeled robots [44]. Additional testing performed on early HOMER base designs has further verified that the IPEC odometry technique is a highly effective method of determining position [45].

This example focuses on systematic uncertainty in the wheel base as it moves in a straight line. For this example, the HOMER velocity profile will exhibit an initial increase phase while it is accelerating to maximum velocity, a constant velocity phase where the robot moves at the maximum velocity, and then a decrease phase where the robot is decelerating to come to a stop. Focusing on the second phase, the dynamics of the system can be modeled as a constant velocity system or

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} x_{1,k+1} \\ x_{2,k+1} \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} = \Phi_k \boldsymbol{x}_k \tag{5.1}$$

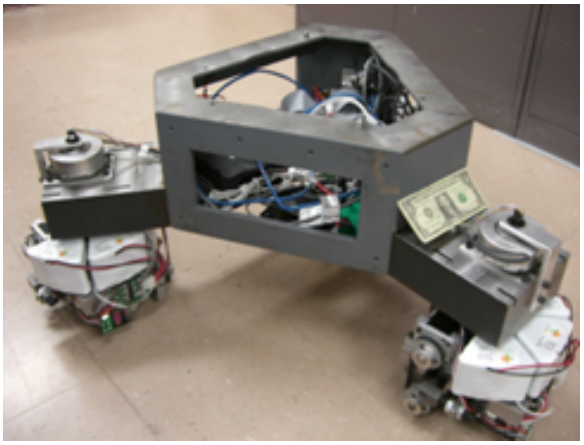


Fig. 34: HOMER Base without the Stewart Platform

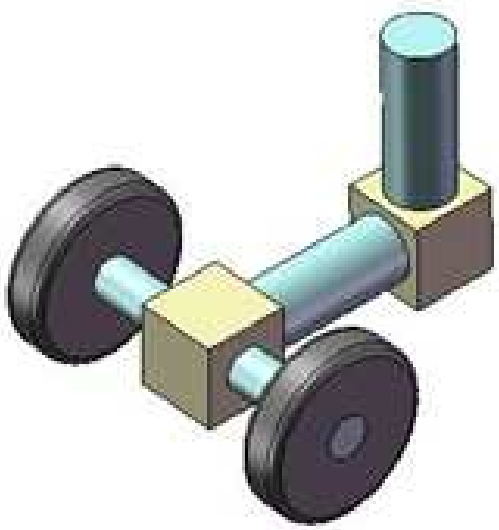


Fig. 35: Conceptual Drawing of an Active Split-offset Castor

where x_1 is the position and x_2 is the velocity. Note that process noise is not being applied to the system since slight variations in the constant velocity assumption can be accounted for in the standard deviation of the velocity estimate. The optical encoder provides an absolute measurement of the angular rotation of the wheel in radians at a rate of 50 Hz. This value can then be multiplied by the radius of the wheel to give an estimate of the position. Thus, the measurement equation is defined as

$$\tilde{y}_k = \frac{x_{1,k}}{r} + v_k, \qquad v_k \sim \mathrm{N}(0, R_k) \tag{5.2}$$

where r is the radius of the wheel. Using the above equations it is possible to define a single state, x_1, a constant dynamic parameter, x_2, and a constant measurement parameter, r. Since EKF versions of the filters will have to be employed, the necessary partial derivatives of the measurement equation are provided by

$$\frac{\partial h}{\partial x_1} = \frac{1}{r}, \qquad \frac{\partial h}{\partial x_2} = 0, \qquad \frac{\partial h}{\partial r} = -\frac{x_1}{r^2} \tag{5.3}$$

where the nonlinear measurement equation, $h(\boldsymbol{x}, t)$, is $\tilde{y}$. Using the above system model a simulation was first performed to analyze the results theoretically prior to applying them to the actual system.

1. Simulation Results

As in previous examples, a Monte Carlo simulation was performed in addition to individual runs to verify assumptions made in each filter. Also as before, the traditional Kalman filter and augmented Kalman filter were tested in addition to the consider Kalman filter to provide a base of comparison. All three filters estimate both the position and velocity as states, but differ in how they observe the uncertainty in the wheel radius. The TKF completely ignores the wheel radius error, the AKF esti-

mates it, and the CKF considers it. The CKF uses the minimum variance approach as described in Chapter III.

The values used to initialize the simulation are given in Table 22. The true values used represent approximate values estimated from a calibration of the actual hardware. Note that the initial standard deviation for the position is zero. This is an actual representation of the system since odometry is a relative measure of position. Thus, the robot position will always be zero relative to itself when the system is updated at a particular instant in time. All cross-covariances between the states and/or parameters are assumed to be initially zero. As mentioned above measurements are taken at a rate of 50 Hz with an assumed standard deviation of 0.45 milliradians. This standard deviation is equivalent to that of a 0.5 encoder count deviation for each wheel. Because encoders typically vary by only ± 1 count for each measurement, this is a reasonable Gaussian approximation.

Table 22: State and Parameter Values Used in the Odometry Hardware Simulation

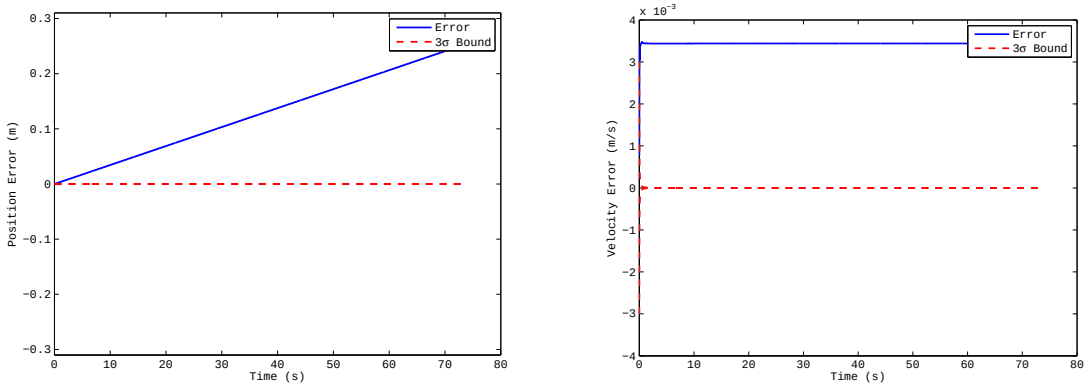
	$x_{1,0}$ (m)	$x_{2,0}$ (m/s)	r (m)
Truth	0	0.058	0.038
Stan. Dev.	0	0.001	0.002

A set of 10000 Monte Carlo trials was evaluated and the results of the final run are shown in Figures 36 - 40 for a single wheel. In this final run the initial velocity is estimated to be 0.0572 m/s and the wheel radius is estimated to be 0.0358. Thus, both values are smaller than the true values causing the positive errors. As seen before the TKF does not accurately account for the parameter error and exhibits larger errors than all of the other filters. Figure 37 shows only the first half second

of the TKF demonstrating how quickly it converges to an incorrect result. The other two filters exhibit similar behavior despite their differences as to whether the wheel radius was estimated or considered. The reason for the similarity is that the wheel radius parameter is unobservable. Therefore, adding it as an additional state in the AKF provides no additional information to the filter. What is clear, however, is that not accounting for the uncertainty in the wheel radius produces inaccurate covariance estimates and larger position errors.

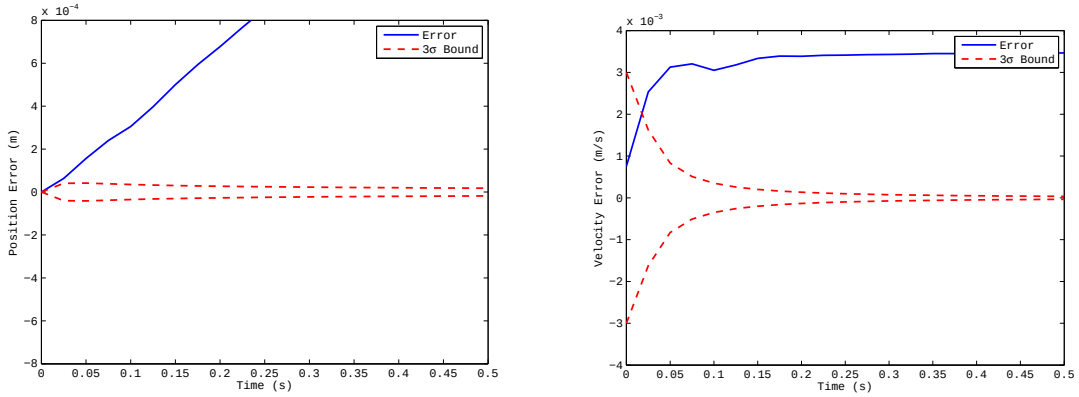
An additional CKF was also tested, where both the velocity and the wheel radius were considered parameters and only the position was estimated. The results of the final run for this filter are shown in Figure 41. Observe that the resulting position estimates using this filter (denoted CKF2) are practically equivalent to those of the AKF and the CKF. Thus, for this particular application as long as both parameters are at a minimum considered better results can be achieved.

Similar to Example IV.2, the results of the Monte Carlo trials are shown in Tables 23 and 24. Although not shown here, the wheel radius estimates for the AKF show comparable statistics to those given for the AKF's position and velocity estimates. These results verify that even with the slight nonlinearities in the measurement model these filters are unbiased. As expected, the TKF true covariance is much larger than the estimated. What is unexpected, however, was that in this case the covariances for the AKF and both CKFs are practically identical. Thus, the results observed in the single run are consistent with the method in general. As a result, in this particular scenario it may be more computationally efficient to consider both the velocity and wheel radius, since estimating these values provides no additional improvement to the position estimate.



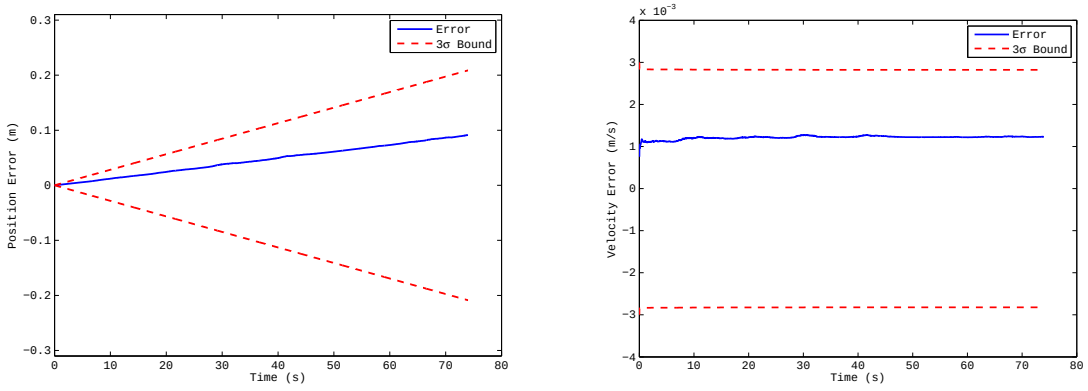
(a) Position (b) Velocity

Fig. 36: State Errors with 3σ Covariance Bounds for the TKF in the Odometry Simulation



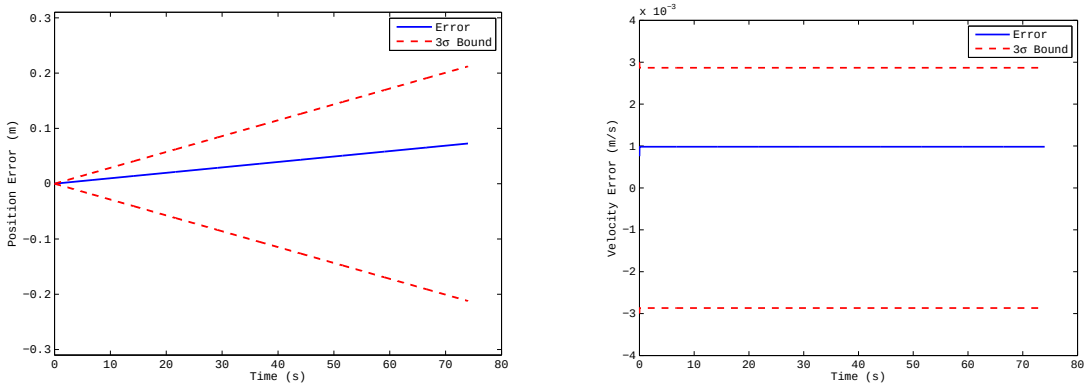
(a) Position (b) Velocity

Fig. 37: Enhanced Initial Position and Velocity Errors with 3σ Covariance Bounds for the TKF



(a) Position (b) Velocity

Fig. 38: State Errors with 3σ Covariance Bounds for the AKF in the Odometry Simulation



(a) Position (b) Velocity

Fig. 39: State Errors with 3σ Covariance Bounds for the CKF in the Odometry Simulation

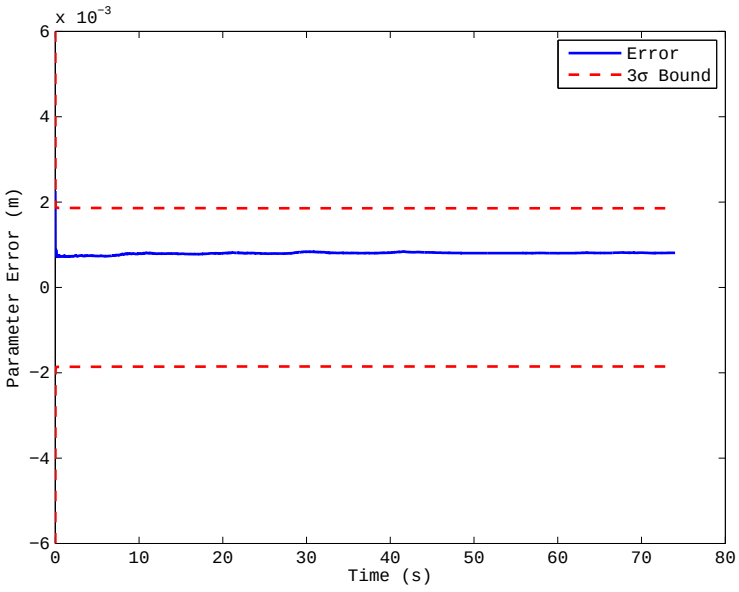


Fig. 40: Wheel Radius Error and 3σ Covariance Bounds for the AKF in the Odometry Simulation

Table 23: Position Error Statistics from both Monte Carlo (MC) and Linear Error Theory (LET) for the Odometry Simulation

	MC Mean (m)	MC Stan. Dev. (mm)	LET Stan. Dev. (mm)
TKF	0.0022	0.226	5.12e-7
AKF	0.0038	0.070	0.070
CKF1	-0.0004	0.070	0.071
CKF2	0.0032	0.069	0.071

Table 24: Velocity Error Statistics from both Monte Carlo (MC) and Linear Error Theory (LET) for the Odometry Simulation

	MC Mean (m/s)	MC Stan. Dev. (m/s)	LET Stan. Dev. (m/s)
TKF	2.95e-5	0.0031	6.92e-9
AKF	5.09e-5	9.46e-4	9.40e-4
CKF1	-5.49e-6	9.39e-4	9.56e-4

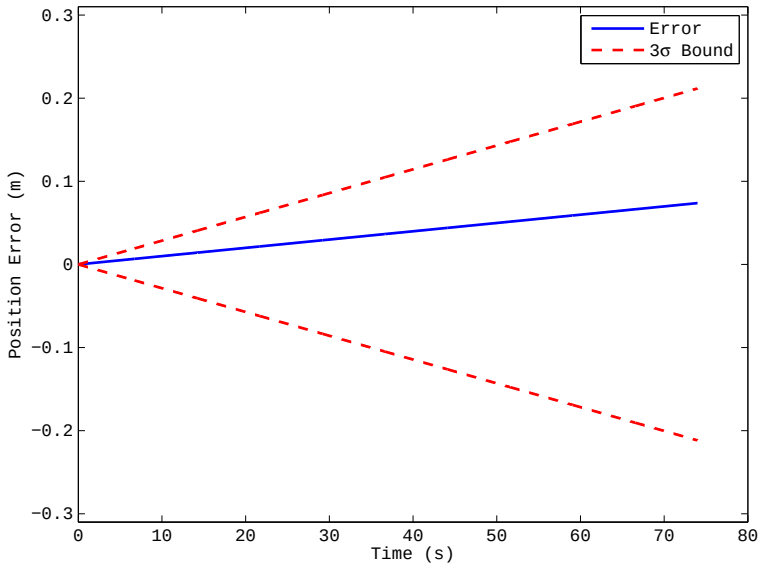


Fig. 41: Position Errors with 3σ Covariance Bounds for the Odometry Simulation Using the Second CKF

2. Hardware Testing Results

Testing of the above simulation was performed on HOMER out at the LASR Laboratory. HOMER was run at 25% of its maximum velocity to provide a large number of data points. Data was collected for all six castor wheels, however, all six wheels exhibited similar behavior. As a result, only the results for one wheel will be described as in the simulations above.

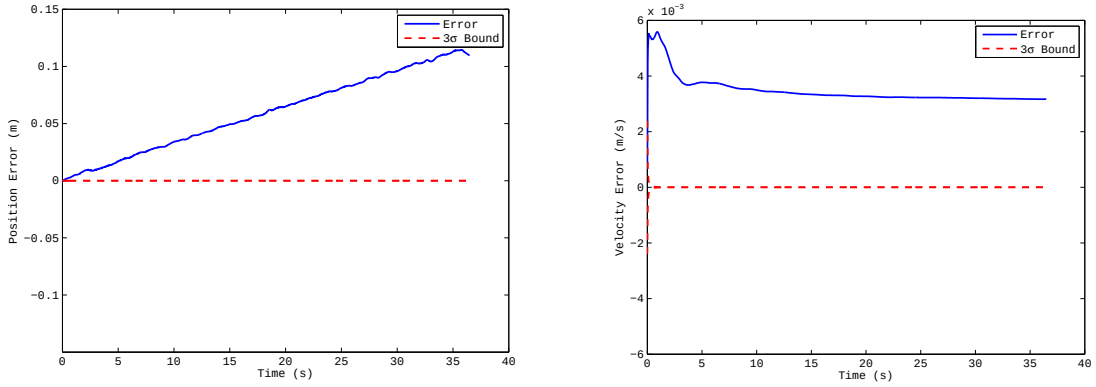
Truth data for this analysis was provided by the VICON motion capture system. The VICON high-speed motion capture system consists of six 16 megapixel cameras and multiple passive retro-reflective markers. The synchronized camera system has sub-millimeter resolution and provides data at a rate of 100 Hz. Fortunately, odometry is a relative measurement of position and calibration to properly orient reference frames between the HOMER base and the VICON system is not required. Since HOMER is moving in a straight line only the magnitude of its translational motion is

required. The assumption is made that all of the wheels have the same radius, because VICON captures the motion of the entire base and not each wheel individually.

Figures 42 - 46 show the results for a single wheel using the same initial estimates as provided by the final run in the simulation testing. Although the TKF, CKF, and second CKF plots exhibit slightly more variation than in the simulation results, the same general trends are observed for each of the filters. The TKF also has more fluctuations in its velocity estimates resulting from the use of real data, but the covariance estimates remain inconsistent and still converge much too rapidly. The major change is in the response of the AKF. Here the unobservability of the wheel radius greatly affects this filter and causes it to go unstable. The *Joseph's form* of the covariance update equation was also used to try and maintain some semblance of stability, but to no avail. Thus, despite the simplicity of this one-dimensional test, the developments from the previous few chapters have been verified. Moreover, it has been shown that in the case of poorly observable or unobservable parameters that it may be possible to still account for their uncertainty using the consider methodology when problems are encountered using either a traditional or augmented Kalman filter.

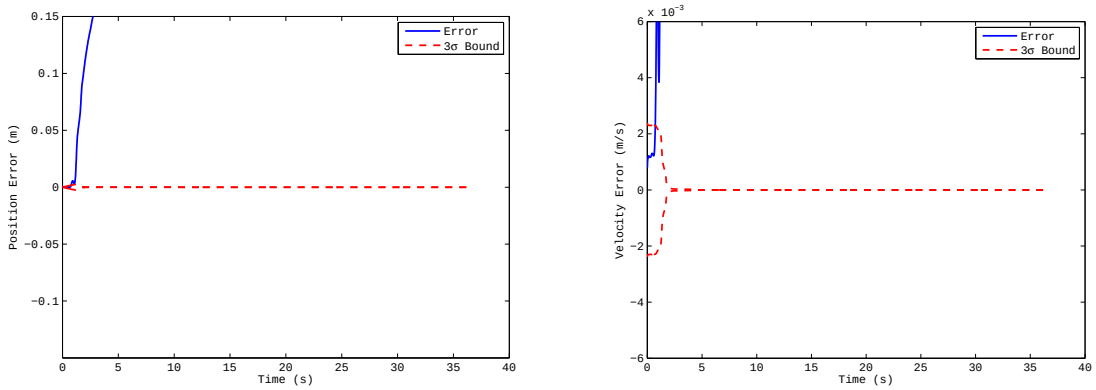
B. Stellar Positioning System

Although the previous hardware example was a relatively simple application, the concepts discussed can also be applied to more complex systems. The Stellar Positioning System (SPS) is an autonomous positioning system to be used by extraterrestrial rovers or as an alternative to GPS here on Earth. For centuries, the stars have been used as a means of position determination for navigation purposes. With today's precise clocks and high quality imaging capabilities, it is possible to accurately determine position using similar methods to those used by early navigators.



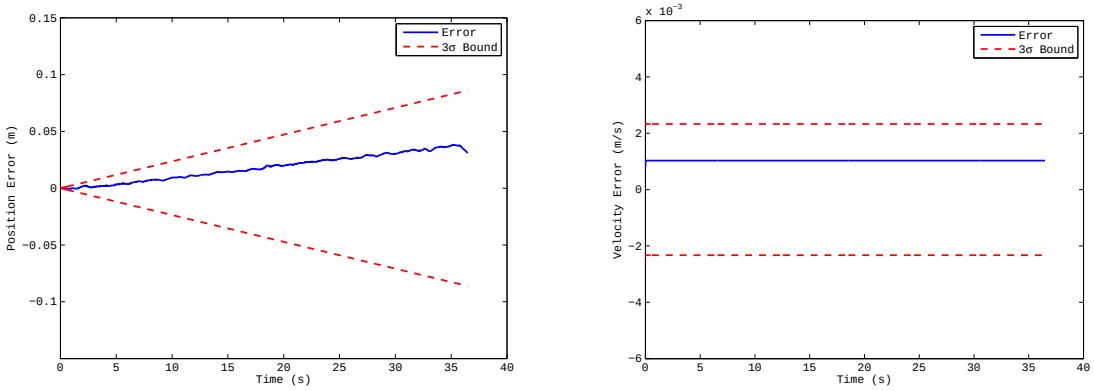
(a) Position (b) Velocity

Fig. 42: State Errors with 3σ Covariance Bounds for the TKF during Odometry Hardware Testing



(a) Position (b) Velocity

Fig. 43: State Errors with 3σ Covariance Bounds for the AKF during Odometry Hardware Testing



(a) Position (b) Velocity

Fig. 44: State Errors with 3σ Covariance Bounds for the CKF during Odometry Hardware Testing

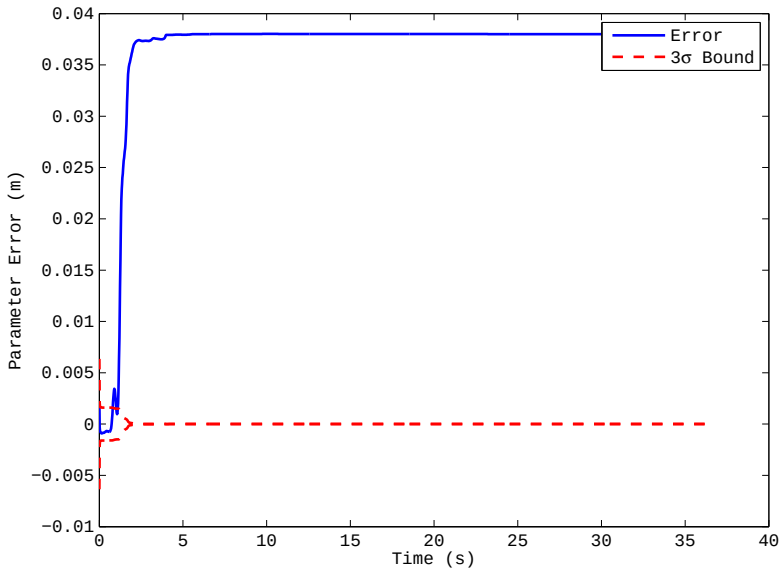


Fig. 45: Wheel Radius Error and 3σ Covariance Bounds for the AKF during Odometry Hardware Testing

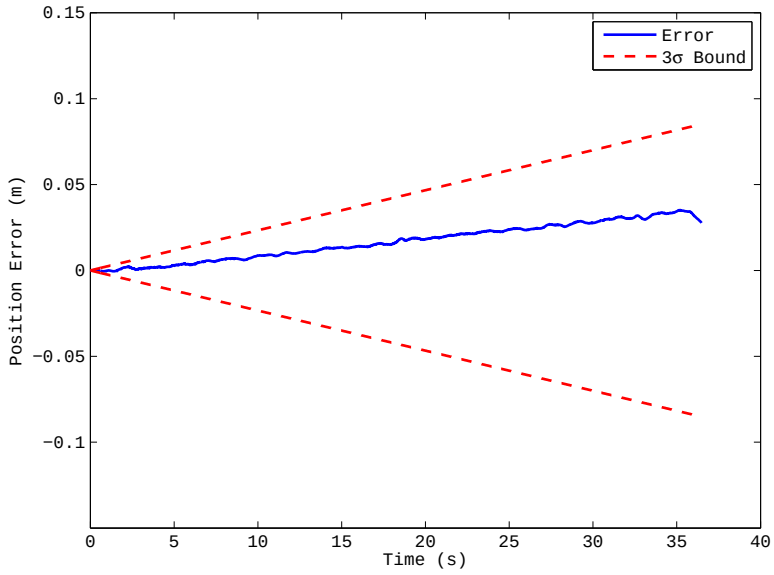


Fig. 46: Position Errors with 3σ Covariance Bounds for Odometry Hardware Testing Using the Second CKF

Taking advantage of these capabilities and highly precise star catalogs, the Stellar Positioning System (SPS) was developed as a modern application of ancient celestial navigation techniques.

While the theory has been developed to determine local latitude and longitude position from interstar angles [46,47], there are additional obstacles to overcome when implementing these concepts in hardware. Since position determination requires a large number of measurements, error creeps into the algorithm from several sources such as the physical environment and the hardware itself. Fortunately, many of these errors can be mitigated using a range of techniques.

One of the major sources of error is the estimation of the attitude between the camera reference frame and the inertial frame being used. This process consists of a number of algorithms including image processing, centroiding, star identification, and attitude estimation. A highly accurate camera model is required for many of

these algorithms as well as for a multiplicative Kalman filter to further refine the attitude estimate [13, 48]. Furthermore, the camera parameters can vary over time due to a number of systematic errors (e.g., temperature fluctuations) and may need to be estimated in real time.

The following sections will discuss in detail the reference frames, hardware, and software components used to implement the SPS. Focus will be placed on the camera model and attitude filter in which the consider methodology will be applied. Experimental results and error analysis are presented based on testing of the completed hardware system.

1. Reference Frames

Several reference frames are employed in the position determination algorithms. The first of these is the inertial reference frame, denoted by I, also called the Earth-Centered Inertial (ECI) frame or the Geocentric Celestial Reference Frame (GCRF). This frame is particularly important because the star reference vectors are oriented in this frame (although centered at the Solar System barycenter, not the Earth). Note that the X-axis of this frame is almost aligned with the vernal equinox, or the ascending node of the geocentric elliptic, and the XY-plane is relatively coplanar with the equatorial plane. To discuss the geographic coordinates on the Earth's surface, it is convenient to introduce a particular epoch to fix the equatorial plane. It is assumed the epoch for this "local inertial reference frame" is at midnight at Greenwich, prior to the first measurement time. This "equator and equinox of date" inertial frame has the important feature that the earth is essentially in pure spin about the Z-axis. Locating this local inertial frame relative to, for example, the GCRF associated with January 1, 2000 requires precision accounting for the precession and nutation of the Earth [47].

The Greenwich reference frame, $\{G\}$, also called the Earth-Centered Earth-Fixed (ECEF) frame or the International Terrestrial Reference Frame (ITRF), describes the location of the Greenwich (or Prime) Meridian that corresponds to a longitude of 0°. The orientation of this frame relative to the inertial frame (Figure 47) is described by the direction cosine matrix (DCM) $\boldsymbol{R}_{G/I}$, which is primarily a function of time, $\Psi(t)$. A simple approximation of this angle would be $\Psi = \omega_e t + \Psi_0$, where ω_e is the Earth rotation rate and Ψ_0 is the angle at some initial time. A more rigorous discussion of the direction cosine matrix (DCM) between the Greenwich and inertial frames, $\boldsymbol{R}_{G/I}$, is presented by Vallado [49].

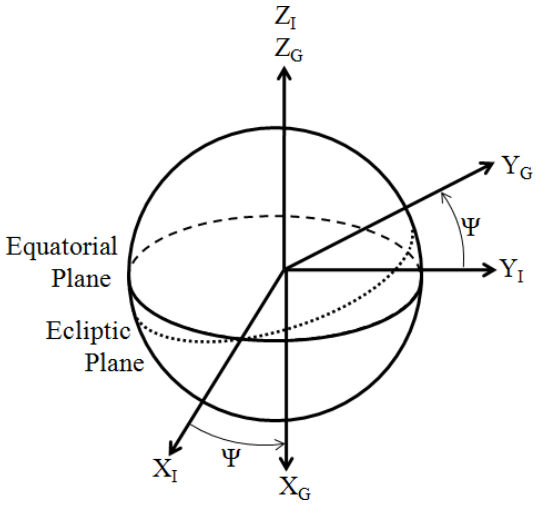


Fig. 47: Inertial $\{I\}$ and Greenwich $\{G\}$ Frames

The orientation of the local frame, $\{L\}$, is described relative to the Greenwich frame via latitude, ϕ, and longitude, λ, angles. These angles are used to construct the DCM between the Local and Greenwich frames, $\boldsymbol{R}_{L/G}$. The X_L and Y_L axes of the local frame are aligned with the standard compass directions South and East, respectively, while the Z_L axis is parallel to the zenith direction. This frame is not traditional, but is chosen to be a right-handed system for convenience. A rotation

about the Z_L axis by a compass angle, θ accounts for the heading angle in the local frame to compass frame, $\{C\}$, direction cosine matrix, $\boldsymbol{R}_{C/L}$. Figure 48 shows the relationship between the three frames. Note that the local frame has been translated to the Earth's surface for visualization only.

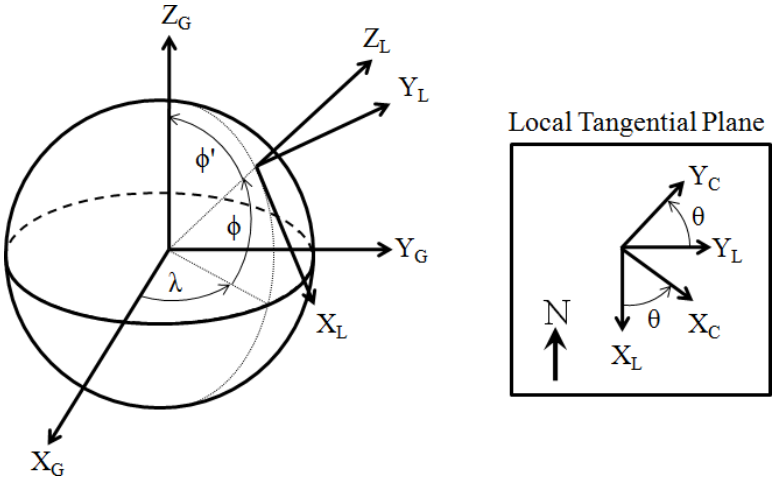


Fig. 48: Greenwich $\{G\}$, Local $\{L\}$, and Compass $\{C\}$ Frames

The next reference frame is the body frame, $\{B\}$, which has an XY-plane that correlates to the plane of the inclinometers. The angle measurements from the inclinometers are used to find the attitude transformation from the compass to the body frame, $\boldsymbol{R}_{B/C}$. The final reference frame is the camera/CCD frame, $\{D\}$. The transformation matrix $\boldsymbol{R}_{D/B}$ provides the transformation from the image plane of the camera to the inclinometer plane of the $\{B\}$ frame. This additional transformation between the camera frame and the inclinometer frame is required due to misalignment errors in the hardware. Together, this set of frames and the DCMs relating the frames are used to estimate the position and/or attitude of the system using the methods presented below.

2. Hardware

The current version of the SPS consists of a high quality astronomical camera, two inclination (i.e., tilt) sensors, a notebook computer, and all necessary cables. The various aspects of the SPS hardware will be discussed in this section.

a. Astronomical Camera

The camera used to validate the SPS theory is a Santa Barbara Instrument Group Model ST-8XME camera. The camera's CCD has dimensions of 1,530 × 1,020 pixels (1.56 Megapixels) and each pixel is 9 microns square. The CCD is thermoelectrically cooled with temperature regulation to within ±0.1°C and has a high quantum efficiency ($\sim 85\%$). The integration time is adjustable from 110 milliseconds to $3,600$ seconds with 10 millisecond resolution. Integration times typically used during testing are between 200 and 500 milliseconds. The camera system includes a USB 1.1 interface for control and image extraction. The download of a full frame image requires approximately 3.7 seconds. Pictures of the camera are shown in Figures 49 and 50.

b. Lens

The camera is equipped to accept C-mount lenses. Manual-focus lenses were chosen to reduce the possible uncertainty associated with an automatic zoom capability. Nikkor lenses were selected as the best quality option given budget constraints. Woodbury *et al.* tested several high quality lenses during the initial phase of the project and found that the 50mm lens with an f/# of 1.2 and a field-of-view (FOV) of 37° provided the best results by returning the most stars with the shortest integration time [48]. This lens was used in all of the testing presented below.

Fig. 49: Top View of Camera (with Inclinometers Attached)

c. Inclinometers

The gravity direction measurement was found using a pair of Wyler Zerotronic $\pm 1^{\circ}$ inclination sensors mounted orthogonal to one another as well as to the optical axis of the camera (see Figures 49 and 50). The sensors transfer the measurements to the computer via a transceiver/converter using a standard serial port. These sensors were chosen for their high precision and range of accuracy. The sensors measure the gravity direction and convert it into a range of angles ($\pm 1^{\circ}$) based on an internal calibration. The current system can measure the inclination angle in approximately 300 milliseconds. The overall orientation of the system requires two measurements, one from each inclinometer, resulting in a set of measurements every 600 milliseconds.

The uncertainty for each sensor is shown in Table 25. In addition to this table, it should be noted that the inclinometers exhibit drift, sensitivity to temperature, and perform averaging on internal measurements taken for longer sampling times. As an example, for a reading sampled at 10 Hz the maximum resolution is 1.29

Fig. 50: Side View of Camera (with Inclinometers Attached)

arcseconds or about 40m on the Earth's surface. To minimize readout error the inclinometers were pointed as close to zenith as possible (Pointing the camera to zenith has the additional benefit of minimizing errors caused by atmospheric refraction for Earth-based measurements). Errors can be further reduced by obtaining multiple measurements from the inclination sensors and filtering the noise.

Table 25: Error Percentages for Wyler Zerotronic ±1° Inclinometers. F.S. - Full Scale Error (full range of sensor) R.O. - Readout Error (error dependent on output value)

Error Type	Amount
Limits of Error in 24 hours @ 20°C	0.017% F.S. + 0.07% R.O.
Limits of Error in 6 months @ 20°C	0.14% F.S. + 0.25% R.O.
Temperature Error / °C	0.04% F.S. + 0.2% R.O.
Resolution (Sampling Time: 0.1s) @ 20°C	±0.0358% F.S.
Resolution (Sampling Time: 1.0s) @ 20°C	±0.0128% F.S.
Resolution (Sampling Time: 10.0s) @ 20°C	±0.00429% F.S.

d. Time

The time is sampled from the Windows XP system clock on the computer. The clock is updated using Network Time Protocol (NTP) from the National Institute of Standards and Technology (NIST) Internet Time Service (ITS). This service provides accuracy to within 20 milliseconds. The Windows XP system clock provides a 15 millisecond granularity. In the current version of the system, no additional interpolation is used to resolve the time to a step size smaller than 15 milliseconds.

Following data collection the time clock was calibrated to coordinated universal time (UTC) using global positioning system (GPS) time measurements. This allowed calculation of both the initial time bias as well as time drift in the computer's internal clock. The internal windows clock was found to get slower by about 1.3 seconds every hour. The initial time bias was found to have a much lower sensitivity than the time drift and was therefore neglected.

3. Software

The theory developed to determine position is summarized in a flow chart as shown in Figure 51. The required measurements are collected by driver programs and then processed using MATLAB code. The images are first searched for stars using centroiding and star-identification algorithms. The vector positions of the stars in the body and inertial frames are the result of this part of the algorithm. The time measurement is then used to construct the Inertial-to-Greenwich Direction Cosine Matrix (DCM), $\boldsymbol{R}_{G/I}$, which is used to evaluate the star position vectors in the Greenwich frame. Next, the inclination measurements are used to estimate the direction of the local gravity vector. The body and Greenwich star location vectors, together with the local gravity direction, are used to solve for the local latitude and

longitude through the inner product solution for position determination [47]. Finally, a GPS receiver is used to collect a "true" measurement of the latitude and longitude for comparison and calibration of the SPS. GPS data points were gathered throughout the data collection process and averaged to find the mean true location. The standard deviation of points surrounding this location was determined to be five meters.

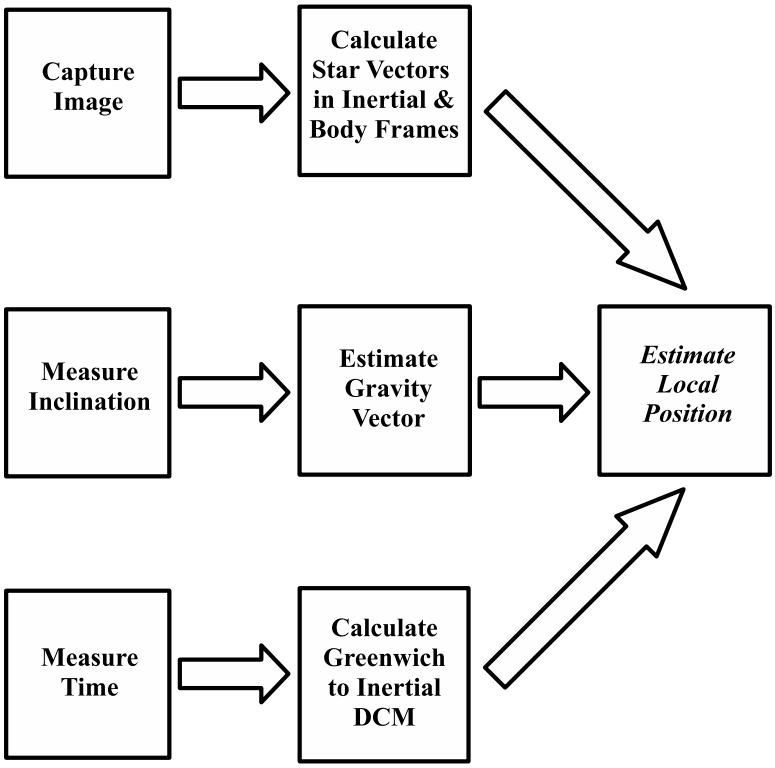


Fig. 51: Basic SPS Local Position Estimation Algorithm

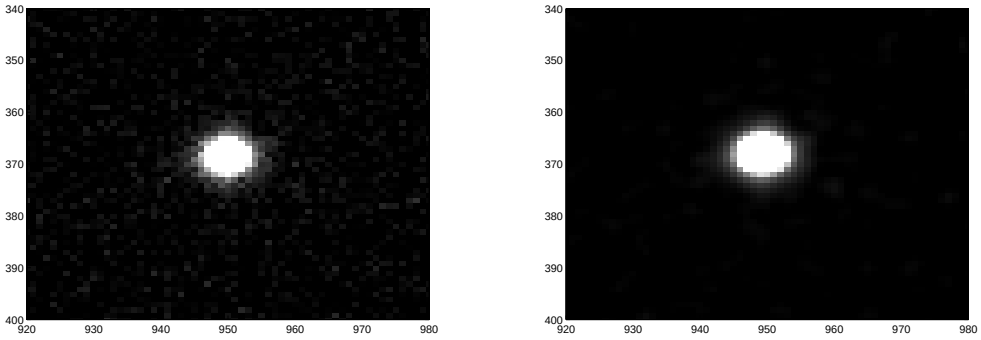
a. Image Processing

Raw images collected from the camera naturally contain noise and artifacts of the imaging system as a whole. Image processing techniques represent a means to remove these artifacts and increase the signal-to-noise ratio (SNR) of the data contained

in the image. For this particular application, image processing is used to increase centroiding accuracy. Centroiding accuracy affects both attitude estimation and star identification, therefore centroid error minimization is critical.

Berry and Burnell discuss multiple prominent image processing techniques including dark frame subtraction, flat-fielding, and taking bias images [50]. While these techniques could be implemented to improve the accuracy of certain imaging systems, a more sophisticated method of artifact removal using level sets was recently developed at Texas A&M [51]. As such, this was the method used during testing.

Often after the artifact removal process, the resultant image will be significantly "pixelated" (see Figure 52(a)). Since the centroiding algorithm assumes that each star centroid has a Gaussian (i.e., normal) distribution, this pixelation can create unwanted effects in the centroid calculation. A Gaussian filter was applied over the entire image following artifact removal to smooth out the image while not significantly changing the centroid locations. The two parameters used by the Gaussian filter were tuned based on simulated numerical tests. The result was solid and well-distributed centroids for the centroiding algorithm (see Figure 52(b)).



(a) Before (b) After

Fig. 52: A Centroid Before and After Using a Gaussian Image Filter

b. Centroiding

The locations of a set of stars, ${}^{D}\boldsymbol{r}$, are needed to calculate the vector positions of the stars in the camera frame. Note that a superscript preceding a star position vector indicates the frame in which the vector is described (D is camera, I is inertial, and G is Greenwich). The primary responsibility of the centroiding algorithm is to best estimate the photocenter for each potential star in the image array.

Although the majority of bright spots in star field images are in fact stars, the possibility of imaging planets and satellites or of having faulty bright pixels (i.e., "hot" pixels) in the array exists. For these reasons, the bright spots are not assumed to be stars until they have been positively identified as such. The most common centroiding approach is using the centroid's center of mass (COM) [50]. Combined with computational efficient algorithms such as run length encoding, the COM is not only very fast, but also very accurate. Additional centroiding methods can be used instead of the COM method (Gaussian-Best-Fit, Peak-Finder, etc.), but these methods only outperformed COM in particular situations [52].

The calculated centroid locations, together with the focal length, f, and CCD offsets, (x_0, y_0), of the camera, provide the components of the star locations in the camera frame, ${}^{D}\boldsymbol{r}_k$, via the collinearity equation:

$$ {}^{D}\boldsymbol{r}_k = \frac{1}{\sqrt{(\hat{x}_k - x_0)^2 + (\hat{y}_k - y_0)^2 + f^2}} \begin{Bmatrix} -(\hat{x}_k - x_0) \\ -(\hat{y}_k - y_0) \\ f \end{Bmatrix} \tag{5.4} $$

To provide an accurate position estimate, this position vector in the camera frame needs to be rotated into the local frame. Typical calculations involve the inclusion of atmospheric refraction as well as diurnal parallax and aberration. Atmospheric refraction is caused by the star light being refracted by the atmosphere. All of the

testing performed was done with the camera pointing at zenith creating negligible effects on the camera vectors. Diurnal aberrations are caused by the velocity of the camera system due to the Earth's rotation. Diurnal parallax results from the camera system not being at the center of the Earth, but on its surface. Both diurnal effects are orders of magnitude smaller than their annual cousins described below and can usually be neglected [53].

c. Star Identification and Attitude Determination

The star identification algorithm represents one of the most vital components of the local position determination algorithm. Without star identification, the attitude of the camera body frame with respect to the inertial frame cannot be determined by the SPS. Similarly, false star identifications will generate erroneous local position output. The star identification algorithm compares the interstar angles of stars in a given star field image to those in a star catalog. The most advanced implementation of this type of star identification is the Pyramid algorithm [54].

The Pyramid algorithm uses a minimum of four (if available) of the ${}^{D}\boldsymbol{r}_k$ vectors from the centroiding algorithm to find a set of (measured) interstar angles. This set of angles is compared to existing values in a star catalog. The angles of the pyramid geometry between the four stars is generally unique, so finding a match to this geometry in the star catalog provides the names of the stars in the image as well as their inertial frame reference vectors, ${}^{I}\boldsymbol{r}_k$, via their established right ascension and declination angles.

Given the position vectors of the image stars in the inertial and camera frames, ${}^{I}\boldsymbol{r}_k$ and ${}^{D}\boldsymbol{r}_k$ respectively, the attitude between the inertial and body frames can be calculated using an attitude determination algorithm. In other words, the optimal solution for $\boldsymbol{R}_{D/I}$ can be found such that ${}^{D}\boldsymbol{r}_k = \boldsymbol{R}_{D/I}{}^{I}\boldsymbol{r}_k$. Among the several existing

algorithms (see Markley and Mortari [55] for a survey), the "Second EStimator of the Optimal Quaternion" (ESOQ-2) [56] was selected for its computational speed and because a flight-tested implementation of it was available. This algorithm finds the optimal quaternion that satisfies both ${}^{D}\boldsymbol{r}_k = \boldsymbol{R}_{D/I}{}^{I}\boldsymbol{r}_k$ and the Wahba optimality criterion [57]. The resulting quaternion can then be used to construct $\boldsymbol{R}_{D/I}$ [36].

To find the inertial directions of the stars, the Tycho-2 catalog [58] was selected. This catalog contains star directions with accuracy of ± 0.6 microarcsec (at J2000) and provides the proper motion with error of ± 0.25 microarcsec/year. This small error results from the multitude of accurate sensor measurements taken from Earth-bound and orbiting observatories over a period of years.

Prior to using the reference star vectors each star vector must be moved into the current time epoch and moved from its reference frame at the barycenter of the solar system to one whose origin is at the center of the Earth (i.e., GCRF). This process involves accounting for the star's proper motion, annual parallax, annual stellar aberration, and sometimes even relativistic light deflection [53, 58]. Proper motion is caused by the fact that the solar system is rotating and stars do not maintain their measured position for all time. The Hipparcos catalog provides methods for this calculation for the Tycho-2 catalog [58]. Annual parallax results from the Hipparcos reference frame not being located at the Earth's center, but at the barycenter of the solar system. Annual stellar aberration causes a change in the star's apparent position with time. Because both the Earth and light itself have finite velocities, the star appears to be in a different location than its measured value. The maximum deviation of 20.6 arcseconds will occur if the star's line-of-sight vector is perpendicular to the Earth's velocity vector. Relativistic light deflection caused by the Sun's gravity well is a much smaller effect than the other three and is neglected in this analysis.

A more thorough description of all of these effects can be found in the Explanatory Supplement to the Astronomical Almanac [53].

Despite these reference frame adjustments, the star catalog error is much smaller than the error associated with the estimated camera vectors, which are based on the image centroiding results and the optical system parameter calibrations. Consequently, the reference vector errors are assumed to be negligible for the purpose of this analysis.

d. Attitude Filtering

The quaternion, $\boldsymbol{q}$, is a commonly used attitude representation since it contains neither parametric nor kinematic singularities. As a result, the quaternion is subject to a normalization constraint which causes problems in an EKF-based application. To get around this, an alternate form of the EKF is used, where the noise is multiplicative as opposed to additive as it has been in all of the previous cases. The result is the aptly named multiplicative Kalman filter (MKF). A derivation of the traditional MKF can be found in Reference 13.

The traditional MKF uses the estimated camera frame star vectors, $\hat{\boldsymbol{b}}$, as the measurements. Using the pinhole camera model described in Eq. (5.4), it is clear that this model also contains three parameters in addition to the states. Thus, the measurement equations can be rewritten as

$$x_c = x_0 + f\frac{C_{11}r_1 + C_{12}r_2 + C_{13}r_3}{C_{31}r_1 + C_{32}r_2 + C_{33}r_3} + v_{x_c} = x_0 + f\frac{D_1}{D_3} + v_{x_c}, \quad v_{x_c} \sim \mathrm{N}(0, \sigma_x^2) \quad (5.5)$$

$$y_c = y_0 + f\frac{C_{21}r_1 + C_{22}r_2 + C_{23}r_3}{C_{31}r_1 + C_{32}r_2 + C_{33}r_3} + v_{y_c} = y_0 + f\frac{D_2}{D_3} + v_{y_c}, \quad v_{y_c} \sim \mathrm{N}(0, \sigma_y^2) \quad (5.6)$$

where (x_c, y_c) is the calculated centroid from the centroiding algorithm, σ_x and σ_y their respective standard deviations, and C_{ij} is the i^{th} row and j^{th} column component of the DCM, C, estimated from the true quaternion, $\boldsymbol{q}$. More complicated models

could have been used, but the pinhole model is a simple and efficient model that accounts for first-order optical effects. Using Eqs. (5.5) and (5.6) allows for the use of either traditional, augmented, or consider Kalman filters in conjunction with the MKF propagation equations, which in turn provides more flexibility and control over the attitude filter being implemented.

The EKF requires partial derivatives, unfortunately, due to the reduced state vector additional, steps must be taken to establish the partial derivatives for the rewritten measurement equations. The true attitude matrix, $C(\boldsymbol{q})$, is related to the estimated *a priori* matrix, $C(\boldsymbol{q}^-)$, by

$$C(\boldsymbol{q}) = C(\boldsymbol{\delta q})C(\hat{\boldsymbol{q}}^-) \tag{5.7}$$

where $\boldsymbol{\partial q}$ is the attitude error. The DCM of the attitude error can be described by the first-order approximation

$$C(\boldsymbol{\delta q}) = I_{3\times 3} - [\boldsymbol{\delta\alpha}\times] \tag{5.8}$$

where $\boldsymbol{\delta\alpha}$ is the small angle approximation for rotation sequences and half of the MKF state vector. Using Eq. (5.8), the associated partial derivatives for Eq. (5.5) are

$$\frac{\partial x_c}{\partial \delta\alpha_1} = -f\frac{D_1 D_2}{D_3^2}, \quad \frac{\partial x_c}{\partial \delta\alpha_2} = f\left(1 + \frac{D_1^2}{D_3^2}\right), \quad \frac{\partial x_c}{\partial \delta\alpha_3} = -f\frac{D_2}{D_3} \tag{5.9}$$

$$\frac{\partial x_c}{\partial x_0} = 1, \quad \frac{\partial x_c}{\partial y_0} = 0, \quad \frac{\partial x_c}{\partial f} = -\frac{D_1}{D_3} \tag{5.10}$$

The partial derivatives for Eq. (5.6) are

$$\frac{\partial y_c}{\partial \delta\alpha_1} = -f\left(1 + \frac{D_2^2}{D_3^2}\right), \quad \frac{\partial y_c}{\partial \delta\alpha_2} = f\frac{D_1 D_2}{D_3^2}, \quad \frac{\partial y_c}{\partial \delta\alpha_3} = f\frac{D_1}{D_3} \tag{5.11}$$

$$\frac{\partial y_c}{\partial x_0} = 0, \quad \frac{\partial y_c}{\partial y_0} = 1, \quad \frac{\partial y_c}{\partial f} = -\frac{D_2}{D_3} \tag{5.12}$$

The measurement partial derivatives with respect to the other three states associated with the gyro drift, $\Delta\hat{\boldsymbol{\beta}}$, are all zero.

The angular velocity estimation used for the SPS varied slightly from the traditional methods, because there are no gyros present in the system. In a typical MKF, a gyro is modeled as

$$\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega} + \boldsymbol{\beta} + \boldsymbol{\eta}_v \tag{5.13}$$

$$\dot{\boldsymbol{\beta}} = \boldsymbol{\eta_u} \tag{5.14}$$

where $\tilde{\boldsymbol{\omega}}$ is the measured angular velocity, $\boldsymbol{\omega}$ is the true angular velocity, $\boldsymbol{\beta}$ is the gyro drift, $\dot{\boldsymbol{\beta}}$ is the gyro drift rate and

$$\boldsymbol{\eta_v} \sim N(0, \sigma_v^2 I_{3\times 3}) \tag{5.15}$$

$$\boldsymbol{\eta_u} \sim N(0, \sigma_u^2 I_{3\times 3}) \tag{5.16}$$

where σ_v and σ_u are the standard deviation of the measurement noise for angular velocity and rate of gyro bias, respectively [59].

In the case of no gyros, however, many of these terms can be ignored. As there is no sensor to measure angular velocity directly, $\tilde{\boldsymbol{\omega}}$ is set to zero. Since Earth's angular velocity is known with great accuracy and is presumed constant, σ_v is a very small number. Thus, to satisfy Eq. (5.13), the initial drift estimate, $\hat{\boldsymbol{\beta}}$, must be set equal to $-\boldsymbol{\omega}$. This leaves σ_u as a tuning parameter. To summarize $\tilde{\boldsymbol{\omega}} = 0$, $\boldsymbol{\beta} = -\boldsymbol{\omega} - \boldsymbol{\eta_v}$, and σ_v is very small. These substitutions allow the filter to include the Earth's rotation in its calculations while not measuring it directly from a gyro. A summary of the traditional MKF used in this application is given in Table 26. Note that

$$\Xi(\boldsymbol{q}) \equiv \begin{bmatrix} q_4 I_{3\times 3} + [\boldsymbol{q}_v\times] \\ -\boldsymbol{q}_v^{\mathrm{T}} \end{bmatrix} \tag{5.17}$$

where q_4 is the scalar portion of the quaternion and $\boldsymbol{q}_v$ is the vector component or

$\boldsymbol{q} = [\boldsymbol{q}_v^{\mathrm{T}} \;\; q_4]^{\mathrm{T}}$. Also,

$$Q(t) = Q = \begin{bmatrix} \sigma_{\alpha_0} I_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & \sigma_{\beta_0} I_{3\times 3} \end{bmatrix} \tag{5.18}$$

e. Position Determination

In order to implement the algorithm outlined in Figure 51, the current time must be sampled and used to construct the inertial to Greenwich DCM, $\boldsymbol{R}_{G/I}$. This coordinate transformation is used to write the star position vectors in the Greenwich frame: ${}^{G}\boldsymbol{r}_k = \boldsymbol{R}_{G/I}{}^{I}\boldsymbol{r}_k$. In accordance with conventions, the transformation is made using the International Astronomical Union (IAU) 2006 resolutions based off of the Celestial Intermediate Origin (CIO) [49]. This procedure correctly orients the GCRF or inertial frame to the ITRF or Greenwich frame.

Two measurements are taken from each inclinometer for each image: one right before the image is taken and one right after. These values are then filtered using a traditional Kalman filter to remove systematic drift effects. The filtered measurements from the inclinometers are then used to estimate the local gravity vector direction using an optimal cones intersection technique [47, 60]. These two components combined with the filtered quaternion are the three pieces of information needed to solve the position determination problem.

4. Experimental Testing Results

On October 14$^{\text{th}}$ 2010, 2000 images were taken over the course of 8 hours using the system described above. The 50mm lens was used with an f/# of 1.8 and an integration time of 500 ms. Additional initial covariance parameters are given in Tables 27 and 28. The initial uncertainty in the camera offset values and focal length

Table 26: Multiplicative Kalman Filter with Pinhole Camera Model Measurements

Gain	$K_k = P_k^- H_k^{\mathrm{T}} \left(H_k P_k^- H_k^{\mathrm{T}} + R\right)^{-1}$ $H_k = f \begin{bmatrix} -\frac{D_1 D_2}{D_3^2} & 1 + \frac{D_1^2}{D_3^2} & -\frac{D_2}{D_3} & 0_{1\times 3} \\ -1 - \frac{D_2^2}{D_3^2} & \frac{D_1 D_2}{D_3^2} & \frac{D_1}{D_3} & 0_{1\times 3} \end{bmatrix}$
Update	$P_k^+ = (I - K_k H_k) P_k^-$ $\Delta\hat{\boldsymbol{x}}_k^+ = K_k \left(\tilde{\boldsymbol{y}} - \boldsymbol{h}_k(\hat{\boldsymbol{x}}_k^-)\right)$ $\Delta\hat{\boldsymbol{x}}_k^+ \equiv \begin{bmatrix} \boldsymbol{\delta}\hat{\boldsymbol{\alpha}}_k^{+^{\mathrm{T}}} & \Delta\hat{\boldsymbol{\beta}}_k^{+^{\mathrm{T}}} \end{bmatrix}^{\mathrm{T}}$ $\boldsymbol{h}_k(\hat{\boldsymbol{x}}_k^-) = \begin{bmatrix} x_c \\ y_c \end{bmatrix} = \begin{bmatrix} x_0 + f\frac{D_1}{D_3} \\ y_0 + f\frac{D_2}{D_3} \end{bmatrix}$ $\hat{\boldsymbol{q}}_k^+ = \hat{\boldsymbol{q}}_k^- + \frac{1}{2}\Xi(\hat{\boldsymbol{q}}_k^-)\boldsymbol{\delta}\hat{\boldsymbol{\alpha}}_k^+$, normalize quaternion $\hat{\boldsymbol{\beta}}_k^+ = \hat{\boldsymbol{\beta}}_k^- + \Delta\hat{\boldsymbol{\beta}}_k^+$
Propagate	$\hat{\boldsymbol{\omega}}(t) = \hat{\boldsymbol{\beta}}(t)$ $\dot{\hat{\boldsymbol{q}}}(t) = \frac{1}{2}\Xi(\hat{\boldsymbol{q}}(t))\hat{\boldsymbol{\omega}}(t)$ $\dot{P}(t) = F(t)P(t) + P(t)F^{\mathrm{T}}(t) + G(t)Q(t)G^{\mathrm{T}}(t)$ $F(t) = \begin{bmatrix} -[\hat{\boldsymbol{\omega}}(t)\times] & -I_{3\times 3} \\ 0_{3\times 3} & 0_{3\times 3} \end{bmatrix}, \quad G(t) = \begin{bmatrix} -I_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & I_{3\times 3} \end{bmatrix}$

are represented by σ_{x_0}, σ_{y_0}, and σ_f respectively. The first 1000 images were used to calibrate the camera, initialize the attitude and inclinometer filters, and allow the system to come to thermal equilibrium with the outside air temperature. Note that typical centroiding algorithms have accuracies as high as a tenth or twentieth of a pixel. The larger measurement covariance used here accounts for additional uncertainties in the camera model not captured by the pinhole representation.

Table 27: State Dynamic Covariance Values Used in SPS Testing

Variables	σ_{α_0}	σ_{β_0} (rad/s^2)	σ_v (rad/s)	σ_u (rad/s^2)
Value	1e-4	1e-5	1e-11	1e-10

Table 28: Parameter and Measurement Covariance Values Used in SPS Testing

Variables	σ_{x_0} (pixels)	σ_{y_0} (pixels)	σ_f (mm)	σ_{x_c} (pixels)	σ_{y_c} (pixels)
Value	0.1	0.1	0.01	1	1

Multiple camera calibration methods are available to date [61, 62], but many of these methods involve using the inner product between two star observations when used with astronomical cameras. The inner product based methods rely on the use of the cosine between two independent star vectors and comparing this result to the same angle between their star catalog inertial vectors. Unfortunately, these angles are typically small resulting in a large number of possible values due to measurement noise and the cosine's insensitivity to small angles. If the sine of the angles from the star vectors' cross product is used instead, greater accuracy can be achieved in the camera calibration at the cost of increased computation time. Using the cross

product technique with the batch least squares method outlined by Tapley *et al.* [8], the camera parameters for the first 1000 images were estimated and the resulting values are shown in Table 29.

Table 29: Estimated Initial Camera Parameter Values Used During SPS Testing

Variables	$\hat{x}_0$ (pixels)	$\hat{y}_0$ (pixels)	$\hat{f}$ (mm)
Value	767.180	506.636	51.57020791

Using the information provided above, an initial analysis of the system was performed with unfiltered attitude estimates. Results of this analysis are shown in Figure 53. 305 points lie within the 1σ bound, 805 in the 2σ bound, and 993 in the 3σ bound. Based on this study, it is clear that these data points represent a behavior that is close to Gaussian. A slight system error still exists, however, since for this to be a perfect 2-dimensional Gaussian additional points should have appeared in both the 1σ and 2σ bounds. Despite this limitation, the unfiltered data represents a very good approximate Gaussian distribution for the testing of multiple attitude filters.

To evaluate the application of the consider methodology on the SPS, a plethora of filters were tested. The typical TKF, AKF, and CKF filters were all tested in addition to all possible combinations of ignoring, estimating, or considering the focal length and/or offsets. The results of the CKF are shown in Figure 54. Notice that the systematic error observed in the unfiltered data is still present as shown by the histogram. The data for the other filters give similar results, but are not shown here. The statistics of the CKF estimates are shown in Table 30 along with all of the other filters. Unfortunately, the current implementation of the SPS requires that

the inclinometer calibration be performed using the estimated attitude vectors. As a result, each filter was calibrated to its own estimates to make a valid comparison.

The first observation is that all of the filters improved over the raw unfiltered results. Also, the largest errors were observed in situations where only the offsets are considered or estimated. This is because the attitude and the offsets are strongly correlated and large errors in the offset estimates will cause large errors in the attitude estimate. Moreover, the focal length tends to vary over time due to temperature fluctuations. The other interesting thing is that considering the focal length produced the best results, but estimating the focal length also produced better results than ignoring all of the parameters. Ignoring both the focal length and the offsets also produced more accurate results than expected. This can be attributed to the high accuracy of the pinhole calibration and that the parameters only varied slightly over the course of the experiment. Similar results to the TKF can also be achieved by applying a moving average to the position estimates using the unfiltered attitude estimates. While the moving average produces reasonable state estimates, its covariance estimates (which are similar to those given by a TKF) have a wide range of variability when compared to other filtering methods.

The unexpected result was that the AKF estimating both focal length and offsets produced worse results than anticipated. By increasing the number of states, the full AKF had more degrees of freedom to try and estimate the best possible result. In doing so, it produced results that were in fact further from the truth. Considering the parameters, however, placed a bound around the possible values for the camera parameters; thereby, in one sense, the possible attitude estimates were constrained. Ultimately, additional calibrations may be necessary to adjust these bounds as the values change with time.

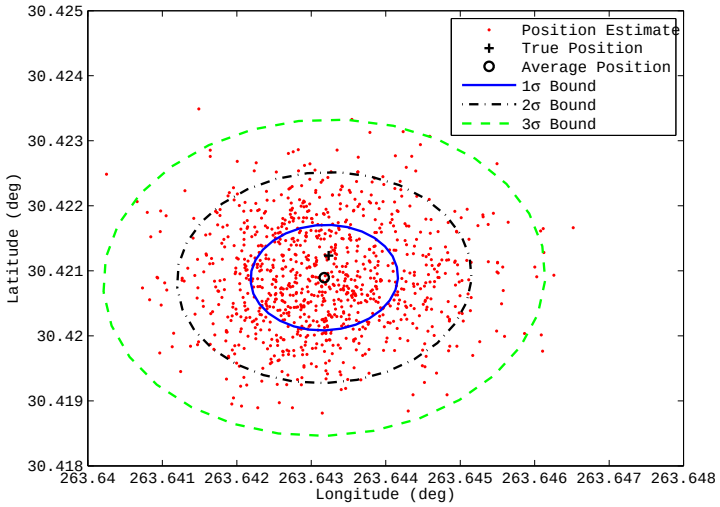


Fig. 53: SPS Unfiltered Data Results

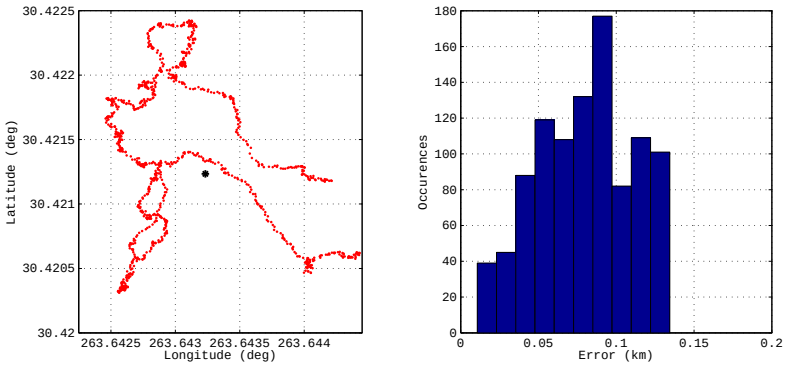


Fig. 54: SPS CKF Data Results

Table 30: Initial Estimated Camera Parameter Values Used in SPS Testing

Filter Description	Maximum Error (km)	Average Error (km)	Standard Deviation (km)
No Filter	0.3207	0.1226	0.0601
Ignore focal length and offsets (TKF)	0.1476	0.0970	0.0372
Estimate focal length and offsets (AKF)	0.1423	0.0939	0.0336
Estimate focal length only (AKF2)	0.1463	0.0901	0.0362
Estimate offsets only (AKF3)	0.1665	0.1336	0.0386
Consider focal length and offsets (CKF)	0.1344	0.0799	0.0310
Estimate focal length and consider offsets (CKF2)	0.1437	0.0894	0.0386
Consider focal length only (CKF3)	0.1341	0.0800	0.0307
Consider offsets only (CKF4)	0.1496	0.0982	0.0378
Consider focal length and estimate offsets (CKF5)	0.1367	0.0808	0.0314

In this particular scenario the CKF outperformed other estimation filters, but this may not always be the case. Since the focal length has a high observability and changes over time due to temperature fluctuations, it is logical to include this parameter as an additional state. On the other hand, the offsets are relatively unobservable and have strong correlations to the attitude estimates. As a result, it makes more sense to consider them and thereby bound their uncertainty. Thus, for a real time application of the MKF using a pinhole camera model, it is sensible to implement CKF2 as the operational filter. This provides a reasonable solution for attitude applications even though the other CKFs provided more accurate results for the Stellar Positioning System.

CHAPTER VI

CONCLUSIONS

In this dissertation, the minimum variance consider Kalman filter (MVCKF) was verified both theoretically and via hardware and software applications to be the best reduced-order filter to date. Consider estimation techniques provide a middle ground between ignoring parameter error and completely accounting for it. Provided that numerical difficulties do not arise in the computation of filter gains, the consider methodology can be used in filtering algorithms to account for the parameter error without directly estimating any or all of the parameters. Furthermore, by accounting for the parameter error, the consider approach provides a rigorous path to improve state estimation through the reduction of both state error and variance for both static and dynamic systems. This truth was verified even for cases where measurement biases are within the bounds of the measurement noise. The consider methodology was shown to produce the same algorithm using least squares, minimum variance, maximum likelihood, and Bayesian measures of estimator optimality for linear Gaussian models. Both the discrete and continuous consider Kalman filters were also shown to be stable, but not asymptotically so.

In addition to the MVCKF, three additional types of reduced-order filters were tested with the presence of parameter errors in the system model. Although parameters are typically augmented to the state vector to allow them to be estimated, often they cannot be included as states due to computational restrictions or poor observability. The traditional Kalman filter (TKF) with process noise was analyzed since it is the predominant method used in today's technical community to account for dynamic parameter uncertainty. The TKF with process noise was able to not

only reduce errors from incorrect parameter estimates, but the predicted covariance estimates were able to bound the error between the state estimates and the truth. The problem, however, is that the process noise covariance matrix requires significant tuning; furthermore, tuning it incorrectly can produce inaccurate state covariance estimates and lead to suboptimal performance.

Another method of state reduction called the minimum variance reduced-order filter (MVROF) was also analyzed. The MVROF is an effective method of accounting for parameter uncertainty in the measurement model using a simple state transformation based off of observer theory. Other works have established that the MVROF should not be used for state vector reduction, because of both biases and computational restrictions [38–40]. The example studied confirmed these results by demonstrating via simulation that the MVROF was unable to account for dynamic parameter uncertainty.

An alternate derivation of the consider Kalman filter was examined as well. The augmented measurement consider Kalman filter (AMCKF) was derived following the theoretical framework for consider analysis described by Tapley *et al.* [8]. Including the parameter covariance in the propagation equations allowed the filter to bound the state error, but an additional tuning parameter, the initial cross-covariance, was introduced. Moreover, in the examples studied, the state estimation error increased with time since the state covariance increased and did not affect the gain calculation. From this analysis, it is apparent that enforcing the constant parameter constraint too early in the system of equations may cause divergent behavior. This is especially true since when this constraint is properly implemented, the resulting AMCKF derivation is equivalent to the MVCKF. As a result, the consider analysis approach presented by Tapley *et al.* cannot be classified as an unbiased, minimum variance approach without imposing certain constraints as described in Chapter IV.

The MVCKF was also tested against other filters wherein the parameter errors were ignored/deleted or where the parameters themselves were estimated by augmenting the state vector. Based on two simple examples, it was clear that for many applications, a failure to account for parameter error can lead to large errors in the state estimates. By estimating the parameters in real time it is possible to account for dynamic parameter uncertainty, but computational and observability concerns must be taken into account and frequently, this ideal approach is infeasible. Hardware testing of the algorithms showed that sometimes numerical instabilities can arise when attempting to estimate weakly observable parameters, but the consider methodology may still be used to account for their uncertainty without numerical instability. Furthermore, if the parameters are poorly observable and strongly correlated with the states, estimating them directly may cause larger deviations from the truth than either ignoring parameter errors or considering them using the methods presented.

As a result of these conclusions, this dissertation has achieved its original goals by:

- Developing a unified framework for reduced-order filters that is theoretically rigorous using present day nomenclature and terminology
- Demonstrating inconsistencies in technical literature on reduced-order filters that can lead to poor real-time implementations of the reduced-order methodology
- Establishing methods to efficiently and effectively reduce high-order state models to account for poorly observable parameters while also decreasing computation costs

- Verification of the methods presented through the use of numerous computer simulations and algorithms implemented to process measurements from hardware experiments

Because of the inconsistencies present in the literature available on reduced-order filters, this dissertation was focused on providing a rigorous framework for reduced-order estimation of dynamic systems. Although not examined here, Total Least Squares represents an avenue for research that may provide additional insight into reduced-order methods that accommodate bilinear systems. Furthermore, the examination of reduced-order filters in this dissertation barely scratched the surface for nonlinear models and it is expected that the developments herein can be extended using Bayesian or particle based filtering methods.

Ultimately, the goal of using a Kalman filter is to provide real-time state estimation to a dynamic system. Many systems should use a combination of both estimated and considered parameters depending on mission requirements and the feasibility of augmenting the uncertain parameters to the state error. Process noise may need to be included as well to account for additional uncertainties such as computer round-off errors or from neglecting higher-order modeling terms. As demonstrated in Chapter IV, the consider methodology can even be used *a priori* to help calculate feasible process noise values if computational restrictions force the user to use only process noise in the on-board real-time implementation. Either way, consider Kalman filtering may not be the ultimate estimation tool, but it provides attractive, versatile, and viable options that can provide improved state estimates and more accurate state covariance estimates for any application involving parameters that are not precisely known.

REFERENCES

[1] Gelb, A., editor, *Applied Optimal Estimation*, The M.I.T. Press, Cambridge, MA, 1974.

[2] Huddle, J. R. and Wismer, D. A., "Degradation of Linear Filter Performance Due to Modeling Error," *IEEE Transactions on Automatic Control*, Vol. 13, No. 4, 1968, pp. 421–423.

[3] Mahalanabis, A. K., "On Minimizing the Divergence in Discrete Filters," *IEEE Transactions on Automatic Control*, Vol. 17, No. 2, 1972, pp. 239–240.

[4] Price, C. F., "An Analysis of the Divergence Problem in the Kalman Filter," *IEEE Transactions on Automatic Control*, Vol. AC-13, December 1968, pp. 699–702.

[5] Duiven, E. M., "Suboptimal Linear Filtering," *Journal of Spacecraft and Rockets*, Vol. 11, No. 3, 1974, pp. 196–198.

[6] Nishimura, T., "On the a priori Information in Sequential Estimation Problems," *IEEE Transactions on Automatic Control*, Vol. 11, No. 2, 1966, pp. 197–204.

[7] Heffes, H., "The Effects of Erroneous Models on the Kalman Filter Response," *IEEE Transactions on Automatic Control*, Vol. 11, No. 3, 1966, pp. 541–543.

[8] Tapley, B., Shutz, B., and Born, G., *Statistical Orbit Determination*, Academic Press, Boston, MA, 2004.

[9] Leondes, C. T., editor, *Theory and Applications of Kalman Filtering*, chap. 4, NATO Advisory Group for Aerospace Research and Design, AGARDograph 139, 1970, pp. 112–116.

[10] Maybeck, P. S., *Stochastic Models, Estimation, and Control*, Vol. 2, Academic Press, New York, NY, 1982.

[11] Aoki, M. and Huddle, J. R., "Estimation of the State Vector of a Linear Stochastic System with a Constrained Estimator," *IEEE Transactions on Automatic Control*, Vol. 12, No. 4, 1967, pp. 432–433.

[12] Hutchinson, C. E., D'Appolito, J. A., and Roy, K. J., "Applications of Minimum Variance Reduced-State Estimators," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-11, No. 5, 1975, pp. 785–794.

[13] Crassidis, J. and Junkins, J., *Optimal Estimation of Dynamic Systems*, Chapman & Hall\CRC Press, Boca Raton, FL, 2004.

[14] Majji, M. and Junkins, J. L., "Total Least Squares Estimation of Dynamical Systems," *2007 AIAA Guidance, Navigation, and Control Conference*, Hilton Head, SC, August 2007.

[15] van Huffel, S. and Vandewalle, J., *The Total Least Squares Problem: Computational Aspects and Analysis*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1991.

[16] van Huffel, S. and Lemmerling, P., editors, *Total Least Squares and Errors-in-variables Modeling*, Springer Books, London, England, 2002.

[17] Bierman, G. J., *Factorization Methods for Discrete Sequential Estimation*, Dover Publications, Inc., Mineola, NY, 1977.

[18] Schmidt, S. F., "Application of State-Space Methods to Navigation Problems," *Advances in Control Systems*, Vol. 3, 1966, pp. 293–340.

[19] Jazwinski, A. H., *Stochastic Processes and Filtering Theory*, Academic Press, Inc., New York, NY, 1970.

[20] Thornton, C. L. and Bierman, G. J., "UDU Covariance Factorization for Kalman Filtering," *Control and Dynamic Systems*, Vol. 16, 1980, pp. 177–248.

[21] Schlee, F. H., Standish, C. J., and Toda, N. F., "Divergence in the Kalman Filter," *AIAA Journal*, Vol. 5, 1967, pp. 1114–1120.

[22] Markley, F. L. and Carpenter, J. R., "Generalized Linear Covariance Analysis," *Journal of the Astronautical Sciences*, Vol. 57, No. 1/2, January–June 2010, pp. 233–260.

[23] Novoselov, R. Y., Herman, S. M., Gadaleta, S. M., and Poore, A. B., "Mitigating the Effects of Residual Biases with Schmidt-Kalman Filter," *2005 International Conference on Information Fusion*, Philadelphia, PA, July 2005.

[24] Ferguson, P. and How, J., "Decentralized Estimation Algorithms for Formation Flying Spacecraft," *2003 AIAA Guidance, Navigation, and Control Conference*, Austin, TX, August 2003.

[25] Parsley, M. P. and Julier, S. J., "The Common State Filter for SLAM," *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nice, France, September 2008.

[26] Kalman, R. and Joseph, P., *Filtering for Stochastic Processes with Applications to Guidance*, Interscience Publishers, New York, NY, 1986.

[27] Franklin, G. F., Powell, J. D., and Workman, M. L., *Digital Control of Dynamic Systems*, Addison-Wesley Publishing Company, Reading, MA, 3rd ed., 1998.

[28] Toda, N., Schlee, F., and Obsharsky, P., "Region of Kalman Filter Convergence for Several Autonomous Navigation Modes," *AIAA Journal*, Vol. 7, No. 4, 1968, pp. 622–627.

[29] Leondes, C. T., editor, *Theory and Applications of Kalman Filtering*, chap. 5, NATO Advisory Group for Aerospace Research and Design, AGARDograph 139, 1970, pp. 112–116.

[30] Slotine, J. and Li, W., *Applied Nonlinear Control*, Prentice Hall, Englewood Cliffs, NJ, 1991.

[31] Miller, J., Konopliv, A., Antreasian, P., Bordi, J., Chesley, S., Helfrich, C., Owen, W., Wang, T., Williams, B., and Yeomans, D., "Determination of Shape, Gravity, and Rotational State of Asteroid 433 Eros," *Icarus*, Vol. 155, No. 1, 2002, pp. 3–17.

[32] Bar-Shalom, Y., Li, X.-R., and Kirubarajan, T., *Estimation with Applications to Tracking and Navigation*, John Wiley & Sons, Inc., New York, NY, 2001.

[33] Thomas, P. C., Parker, J. W., McFadden, L. A., Russell, C. T., Stern, S. A., Sykes, M. V., and Young, E. F., "Differentiation of the Asteroid Ceres as Revealed by its Shape," *Nature*, Vol. 437, 2005, pp. 224–226.

[34] Carry, B., Dumas, C., Fulchignoni, M., Merline, W. J., Berthier, J., Hestroffer, D., Fusco, T., and Tamblyn, P., "Near-Infrared Mapping and Physical Properties of the Dwarf-Planet Ceres," *Astronomy & Astrophysics*, Vol. 478, 2007, pp. 235–244.

[35] Chamberlain, M. A., Sykes, M. V., and Esquerdo, G. A., "Ceres Lightcurve Analysis - Period Determination," *Icarus*, Vol. 188, 2007, pp. 451–456.

[36] Schaub, H. and Junkins, J. L., *Analytical Mechanics of Space Systems*, AIAA Education Series, Reston, VA, 2003.

[37] Stark, H. and Woods, J. W., *Probability and Random Process with Applications to Signal Processing*, Prentice Hall, Upper Saddle River, NJ, 3rd ed., 2002.

[38] Asher, R. B., Herring, K. D., and Ryles, J. C., "Bias, Variance, and Estimation Error in Reduced Order Filters," *Automatica*, Vol. 12, 1976, pp. 589–600.

[39] Setterlund, R. H., "New Insights into Minimum-Variance Reduced-order Filters," *Journal of Guidance, Control, and Dynamics*, Vol. 11, No. 6, 1988, pp. 495–499.

[40] Kerr, T. H., "The Proper Computation of the Matrix Pseudoinverse and its Impact in MVRO Filtering," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-21, No. 5, 1985, pp. 711–724.

[41] Davis, J. J., Doebbler, J., Daugherty, K. J., Junkins, J. L., and Valasek, J., "Aerospace Vehicle Motion Emulation Using Omni-directional Mobile Platform," *2007 AIAA Guidance, Navigation, and Control Conference*, Hilton Head, SC, August 2007.

[42] Doebbler, J., Davis, J. J., Valasek, J., and Junkins, J. L., "Mobile Robotic System for Ground Testing of Multi-Spacecraft Proximity Operations," *2008 AIAA Modeling and Simulation Technologies Conference*, Honolulu, HI, August 2008.

[43] Borenstein, J., Everett, H., Feng, L., and Wehe, D., "Mobile Robot Positioning: Sensors and Techniques," *Journal of Robotic Systems*, Vol. 14, No. 4, 1997, pp. 231–249.

[44] Borenstein, J., Everett, H., and Feng, L., *Navigating Mobile Robots: Sensors and Techniques*, AK Peters Ltd, Wellesley, MA, 1996.

[45] Doebbler, J., Davis, J. J., Junkins, J. L., and Valasek, J., "Odometry and Calibration Methods for Multi-Castor Vehicles," *2008 IEEE International Conference on Robotics and Automation*, Pasadena, CA, May 2008.

[46] Swanzy, M. J., "Analysis and Demonstration: A Proof-of-concept Compass Star Tracker," M.S. Thesis, Texas A&M University, 2005.

[47] Parish, J., Parish, A., Swanzy, M., Woodbury, D., Mortari, D., and Junkins, J. L., "The Stellar Positioning System (Part I): An Autonomous Position Determination Solution," *Navigation*, Vol. 57, No. 1, 2010, pp. 1–12.

[48] Woodbury, D., Parish, J., Parish, A., Swanzy, M., Denton, R., Mortari, D., and Junkins, J. L., "The Stellar Positioning System (Part II): Improving Accuracy During Implementation," *Navigation*, Vol. 57, No. 1, 2010, pp. 13–24.

[49] Vallado, D. A., *Foundations of Astrodynamics and Applications*, Kluwer Academic Publishers, El Segundo, CA, 2nd ed., 2001.

[50] Berry, R. and Burnell, J., *The Handbook of Astronomical Image Processing*, Willmann-Bell, Richmond, VA, 2nd ed., 2005.

[51] Flewelling, B. R. and Mortari, D., "Information Theoretic Weighting for Robust Star Centroiding," *2010 AAS Space Flight Mechanics Meeting*, San Diego, CA, February 2010.

[52] Mortari, D., Bruccoleri, C., La Rosa, S., and Junkins, J. L., "CCD Data Processing Improvements," *2002 International Conference on Dynamics and Control of*

Systems and Structures in Space, King College, Cambridge, England, July 14–18, 2002.

[53] Seidelmann, P. K., editor, *Explanatory Supplement to the Astronomical Almanac*, University Science Books, Sausalito, CA, 1992.

[54] Mortari, D., Samaan, M. A., Bruccoleri, C., and Junkins, J. L., "The Pyramid Star Identification Technique," *Navigation*, Vol. 51, No. 3, Fall 2004, pp. 171–183.

[55] Markley, F. L. and Mortari, D., "Quaternion Attitude Estimation Using Vector Observations," *Journal of the Astronautical Sciences*, Vol. 48, No. 2/3, April–September 2000, pp. 359–380.

[56] Mortari, D., "Second Estimator of the Optimal Quaternion," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 5, September–October 2000, pp. 885–888.

[57] Wahba, G., "A Least-Squares Estimate of Satellite Attitude," *SIAM Review*, Vol. 7, No. 3, 1965, pp. 409.

[58] Perryman, M. A. C. and ESA, editors, *The HIPPARCOS and TYCHO Catalogues. Astrometric and photometric star catalogues derived from the ESA HIPPARCOS Space Astrometry Mission*, Vol. 1200 of *ESA Special Publication*, 1997.

[59] Farrenkopf, R., "Analytic Steady State Accuracy Solutions for Two Common Spacecraft Attitude Estimators," *Journal of Guidance and Control*, Vol. 1, No. 4, July - August 1978, pp. 282 – 284.

[60] Mortari, D. and Singla, P., "Optimal Cones Intersection Technique," *ACTA Astronautica*, Vol. 59, No. 6, September 2006, pp. 474–482.

[61] Zhang, Z., "A Flexible New Technique for Camera Calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 11, 2000, pp. 1330–1334.

[62] Samaan, M. A., Mortari, D., and Junkins, J. L., "Non–dimensional Star Identification for Un–calibrated Star Cameras," *Journal of the Astronautical Sciences*, Vol. 54, No. 1, January–March 2006, pp. 95–111.

APPENDIX A

MATRIX TRACE CALCULUS

Taking the partial derivative of the trace of a matrix is used frequently in Kalman filter derivations to find the minimum variance solution for the gain. Some useful derivatives are given by

$$\frac{\partial}{\partial K}\mathrm{tr}\left(KA\right) = A^{\mathrm{T}} \tag{A.1}$$

$$\frac{\partial}{\partial K}\mathrm{tr}\left(AK^{\mathrm{T}}\right) = A \tag{A.2}$$

$$\frac{\partial}{\partial K}\mathrm{tr}\left(KAK^{\mathrm{T}}\right) = KA^{\mathrm{T}} + KA \tag{A.3}$$

where K and A are two arbitrary matrices satisfying matrix multiplication rules.

APPENDIX B

SHERMAN-MORRISON-WOODBURY LEMMA

The Sherman-Morrison-Woodbury matrix inversion lemma is used in Kalman filter derivations to define the gain in terms of the propagated covariances instead of the updated covariances. Let

$$F = [A + BCD]^{-1} \tag{B.1}$$

where F, A, B, C, and D are all arbitrary matrices satisfying matrix multiplication rules. Assuming all inverses exist it can be shown that

$$F = A^{-1} - A^{-1}B\left(DA^{-1}B + C^{-1}\right)^{-1}DA^{-1} \tag{B.2}$$

Proofs of this can be found in multiple locations, one such being in Crassidis and Junkins [13].

VITA

Drew Patton Woodbury was born in Washington, D.C. He received his Bachelor of Science degree in Mechanical Engineering from the Massachusetts Institute of Technology, Cambridge, Massachusetts in June 2003. Upon graduation, Drew received his commission from the United States Air Force and was assigned as a Space and Missile Operations Officer. After serving a tour at Malmstrom AFB, Great Falls, Montana as a Deputy ICBM Combat Crew Commander Instructor, Drew left the military to pursue higher education. He began his graduate study at Texas A&M University in August 2007 as a Bradley Fellow under Dr. John L. Junkins. In 2008, Drew was named as a National Defense Science & Engineering Graduate (NDSEG) Fellow. His research interests focused on reduced-order system estimation methods, star tracking sensor calibration and applications, and space situational awareness. Upon graduation in 2011, Drew accepted a job from Sandia National Laboratories where he will serve as a Senior Member of Technical Staff in the Mission Systems Engineering Group (5572).

Drew Patton Woodbury's email address is dwoodbury@alum.mit.edu. He can be reached at:

Department of Aerospace Engineering

c/o Dr. John L. Junkins

Texas A&M University

H.R. Bright Building, Ross Street - TAMU 3141

College Station, TX 77843-3141